U.S. Department
of Transportation
**National Highway
Traffic Safety
Administration**

DOT HS 808 282
DOT-VNTSC-NHTSA-94-5

Final Report
August 1995

Transformation of Nine-Accelerometer-Package (NAP) Data for Replicating Headpart Kinematics and Dynamic Loading

Frank DiMasi

Research and
Special Programs
Administration
Volpe National
Transportation Systems Center
Cambridge, MA 02142-1093

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 1995	Final Report July 1992 - August 1994

4. TITLE AND SUBTITLE
TRANSFORMATION OF NINE-ACCELEROMETER-PACKAGE (NAP) DATA FOR REPLICATING HEADPART KINEMATICS AND DYNAMIC LOADING

5. FUNDING NUMBERS
HS476/S4004

6. AUTHOR(S)
Frank P. DiMasi

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Department of Transportation
Research and Special Projects Administration
Volpe Center
Cambridge, MA 02142

8. PERFORMING ORGANIZATION REPORT NUMBER
DOT-VNTSC-NHTSA-94-5

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Department of Transportation
National Highway Traffic Safety Administration
Office of Research and Development
Washington, DC 20590

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
DOT HS 808 282

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
This document is available to the public through the National Technical Information Service, Springfield, VA 22161

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This report describes a procedure and associated software package for transforming time-varying kinematic quantities computed in a rotating-decelerating reference frame, to a fixed coordinate system. Time-varying kinematic quantities are computed, based on accelerometer measurements made with a 3-2-2-2 nine accelerometer package (NAP) located in the headpart of an anthropometric test device, and transformed to fixed coordinates in the form of generalized six-degree-of-freedom velocity time histories at the headpart center of gravity. An algorithm is also included to compensate for cross products of angular velocity and centripetal accelerations arising from the non-coplanar geometry of the NAP.

Angular and translational velocity components are then applied as velocity boundary conditions, to the center of gravity of finite-element representation of a rigid skull-headpart component of an anatomic brain model in order to replicate the generalized six-degree-of-freedom kinematics, and corresponding inertial loads produced by these kinematics. These efforts are part of a larger program to develop tools and methods aimed at providing an improved head injury criteria for use in automobile crash testing.

Principal elements of this report include (a) principles of the transformation process; (b) use of the software and a typical interactive session; (c) comparisons with crash test film data; (d) use of simulated accelerometer data for validating the transformation process; (e) characterization of typical errors due to non-coplanar geometry of the 3-2-2-2 array; and (f) a compensation algorithm for eliminating uncertainties associated with non-coplanar geometry. A program listing is also included.

14. SUBJECT TERMS
Coordinate Transformations, Biomechanics, Head Impact, Dynamic Head Loading, Head Trauma, Anthropometric Test Device, Rotating Coordinate Systems, Accelerometer Array, Nine Accelerometer Package

15. NUMBER OF PAGES
184

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	

PREFACE

The work described in this report was sponsored by the US DOT Office of Research and Development of the National Highway Traffic Safety Administration (NHTSA) under Project Plan Agreement HS-76. From the beginning of this effort, Dr. Rolf Eppinger, Chief of the NHTSA's Biomechanics Division, provided critical support and direction for this work, and in the latter stages of the work, Dr. Faris Bandak provided additional direction and support.

The work described herein is part of a larger effort to develop tools and methods to provide an improved head injury criteria for use in automobile crash testing. The current head injury criteria is based on translational accelerations measured at the center of gravity of a dummy headpart, and does not consider rotational kinematics which have been shown to have a strong effect on neurological impairment due to strains developed in the soft tissue of the brain (see Reference 2). Finite element modeling of the skull and brain, and replicating combined translational and rotational dynamic loading sustained by the head, are among the efforts supported by NHTSA's Biomechanics Division to reach these goals.

Specifically, this report describes the procedure and associated software for computing angular and translational headpart kinematics with respect to inertial coordinates, by transforming non-inertial accelerometer data measured in a 3-2-2-2 nine-accelerometer array located in the dummy headpart. Generalized six-degree-of-freedom angular and translational velocity components may then be applied as boundary conditions, to an anatomic model of the skull and brain. This process involves the transformation of time-varying vector quantities from a body coordinate system undegoing generalized rotation, to a fixed reference frame. Principal elements of this report include: (a) principles of the transformation process; (b) use of the software and a typical interactive session; (c) comparisons with crash test film data; (d) use of simulated accelerometer data for validating the transformation process; (e) characterization of typical errors due to non-coplanar geometry of the 3-2-2-2 array; and (f) a compensation algorithm for eliminating uncertainties associated with non-coplanar geometry. A program listing is also included.

The basic transformation algorithm described herein was developed by Dr. Weinstock of the Volpe Center. Dr. Tong of the Hong Kong University for Science and Technology was instrumental in developing the compensation algorithm for non-coplanar geometry.

METRIC/ENGLISH CONVERSION FACTORS

ENGLISH TO METRIC

LENGTH (APPROXIMATE)

1 inch (in.) = 2.5 centimeters (cm)
1 foot (ft) = 30 centimeters (cm)
1 yard (yd) = 0.9 meter (m)
1 mile (mi) = 1.6 kilometers (km)

AREA (APPROXIMATE)

1 square inch (sq in, in^2) = 6.5 square centimeters (cm^2)
1 square foot (sq ft, ft^2) = 0.09 square meter (m^2)
1 square yard (sq yd, yd^2) = 0.8 square meter (m^2)
1 square mile (sq mi, mi^2) = 2.6 square kilometers (km^2)
1 acre = 0.4 hectares (he) = 4,000 square meters (m^2)

MASS - WEIGHT (APPROXIMATE)

1 ounce (oz) = 28 grams (gr)
1 pound (lb) = .45 kilogram (kg)
1 short ton = 2,000 pounds (lb) = 0.9 tonne (t)

VOLUME (APPROXIMATE)

1 teaspoon (tsp) = 5 milliliters (ml)
1 tablespoon (tbsp) = 15 milliliters (ml)
1 fluid ounce (fl oz) = 30 milliliters (ml)
1 cup (c) = 0.24 liter (l)
1 pint (pt) = 0.47 liter (l)
1 quart (qt) = 0.96 liter (l)
1 gallon (gal) = 3.8 liters (l)
1 cubic foot (cu ft, ft^3) = 0.03 cubic meter (m^3)
1 cubic yard (cu yd, yd^3) = 0.76 cubic meter (m^3)

TEMPERATURE (EXACT)

[(x - 32) (5/9)] °F = y°C

METRIC TO ENGLISH

LENGTH (APPROXIMATE)

1 millimeter (mm) = 0.04 inch (in)
1 centimeter (cm) = 0.4 inch (in)
1 meter (m) = 3.3 feet (ft)
1 meter (m) = 1.1 yards (yd)
1 kilometer (km) = 0.6 mile (mi)

AREA (APPROXIMATE)

1 square centimeter (cm^2) = 0.16 square inch (sq in, in^2)
1 square meter (m^2) = 1.2 square yards (sq yd, yd^2)
1 square kilometer (kn^2) = 0.4 square mile (sq mi, mi^2)
1 hectare (he) = 10,000 square meters (m^2) = 2.5 acres

MASS - WEIGHT (APPROXIMATE)

1 gram (gr) = 0.036 ounce (oz)
1 kilogram (kg) = 2.2 pounds (lb)
1 tonne (t) = 1,000 kilograms (kg) = 1.1 short tons

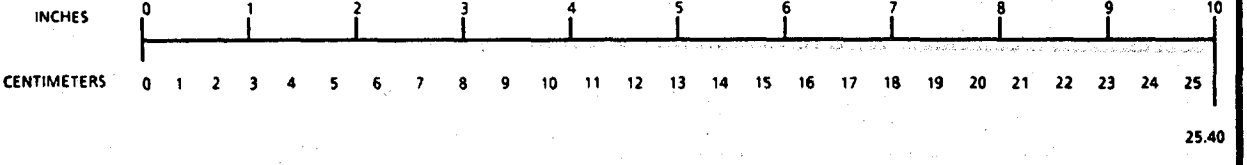
VOLUME (APPROXIMATE)

1 milliliter (ml) = 0.03 fluid ounce (fl oz)
1 liter (l) = 2.1 pints (pt)
1 liter (l) = 1.06 quarts (qt)
1 liter (l) = 0.26 gallon (gal)
1 cubic meter (m^3) = 36 cubic feet (cu ft, ft^3)
1 cubic meter (m^3) = 1.3 cubic yards (cu yd, yd^3)

TEMPERATURE (EXACT)

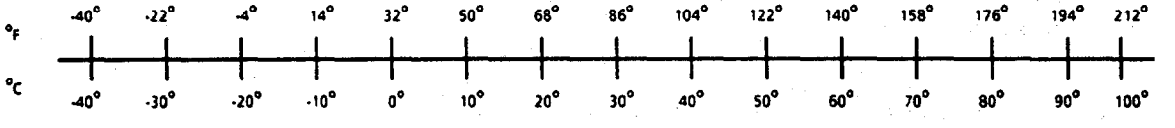
[(9/5) y + 32] °C = x °F

QUICK INCH-CENTIMETER LENGTH CONVERSION



25.40

QUICK FAHRENHEIT-CELSIUS TEMPERATURE CONVERSION



For more exact and or other conversion factors, see NBS Miscellaneous Publication 286, Units of Weights and Measures. Price $2.50. SD Catalog No. C13 10286.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont.)

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF TABLES

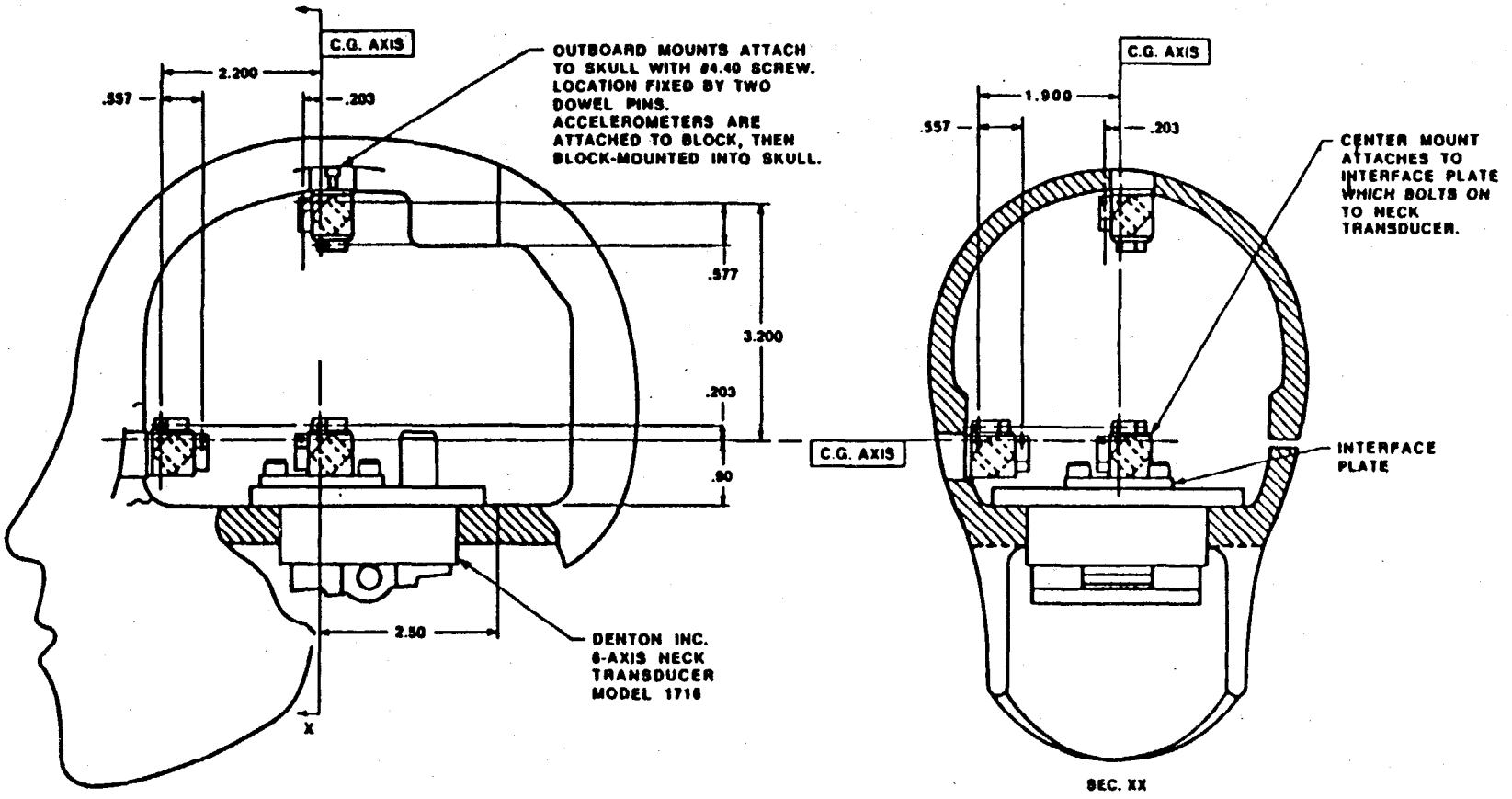
1. INTRODUCTION

Head injuries sustained in the automobile collision environment are currently evaluated according to the Head Injury Criterion (HIC) specified in Federal Motor Vehicle Safety Standard (FMVSS) 208, which is based upon the vector sum of translational accelerations measured at the center of gravity of an anthropometric dummy headpart. This criterion, which evolved from the Wayne State Tolerance Curve [1], relates the probability of head injury to temporal and amplitude features of the resultant acceleration time history measured at the head's center of gravity, and provides a quantitative indication of the potential for serious head injury.

Although HIC has been shown to be indicative of many forms of head injury, questions have been raised about its ability to predict diffuse injuries to soft brain tissue, which do not result in externally identifiable lesions. In recent years, experimental and patho-physiological studies have strongly suggested that axonal damage is related to the magnitude of strain experienced by the imposed dynamic loading, and that these strains are strongly related to imposed angular kinematics. Bridging vein disruption is also believed to be the cause of many brain hematomas. A detailed discussion of the influence of angular kinematics and its potential effects on induced strains is contained in [2].

To begin to understand the relationship between angular kinematics and strains induced in soft brain tissue, a three-dimensional anatomic brain model was constructed [3] in 1991 and various impacts with upper interior surfaces were simulated by launching the model into upper interior contact surfaces with a specified initial velocity. While these simulations can produce representative responses for short impact durations, they also require detailed modeling of the interior structure and neglect significant effects of forces and torques occurring at the head-neck interface. To effectively use three-dimensional anatomic models, a more general capability is required for replicating and applying the generalized six-degree-of-freedom dynamic loads measured in the actual collision test environment.

This report describes the development and application of coordinate transformation software for transforming non-inertial accelerometer measurements made in a dummy headpart using a 3-2-2-2 Nine Accelerometer Package (NAP) array, to a non-rotating, inertial coordinate system. Accelerometer measurement locations within the dummy headpart are illustrated in the drawing of Figure 1-1, for the 3-2-2-2 NAP configuration, and the ideal or coplanar arrangement of accelerometers is illustrated in Figure 1-2. In the 3-2-2-2 NAP configuration, three accelerometers are located at the center of gravity (CG) and two accelerometers are displaced along each principal body axis with sense axes orthogonal to that axis and aligned with the remaining principal body axes.



NOTES

1 *DIMENSIONS GIVEN ARE FOR ENDEVCO 7264-2000 ACCELEROMETERS ONLY.*

2 THE TOP, FRONT AND LEFT SIDE MOUNTS ARE INTERCHANGEABLE.

3 POSITION ACCURACY OF BLOCKS IN HEAD 1.005

4 POSITION ACCURACY OF SEISMIC MASS WITHIN ACCELEROMETER ±.03 (PER ENDEVCO SPEC.)

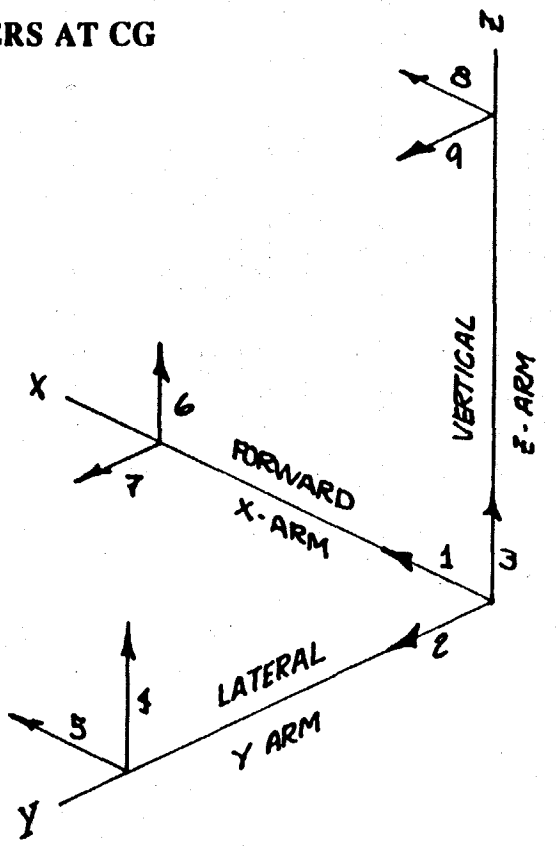
5 CENTER OF GRAVITY (C.G.) LOCATIONS PER G.M. DRAWINGS 78051-61 AND 78051-338 FOR HYBRID III TEST DUMMY HEAD.

FIGURE. 1-1. ACCELEROMETER MEASUREMENT LOCATIONS WITHIN DUMMY HEADPART 3-2-2-2 NAP CONFIGURATION

NINE ACCELEROMETER PACKAGE:

LOCATION OF ACCELEROMETERS AT CG

AND ON X, Y AND Z ARMS:



NUMBERS CORRESPOND TO THE ORDER IN WHICH FILES ARE READ

FIGURE 1-2. IDEAL OR COPLANAR ARRANGEMENT OF ACCELEROMETERS 3-2-2-2 NAP CONFIGURATION

The x, y, and z arms are respectively oriented along the dummy headpart's forward (anterior-posterior), lateral and vertical (superior-inferior) axes, and generally correspond with vehicle axis labeling. The position of accelerometers in the array is very important in the transformation process and for the discussion which follows. The relationship between accelerometer measurements and its position in the NAP array, is indicated in Figure 1-2 (this will be discussed further).

Using the procedure described herein, the three translational and rotational velocities at the headpart center of gravity are derived from the NAP data to describe the generalized six-degree-of-freedom (DOF) motion of the head in terms of velocity time histories at the center of gravity. The generalized six-degree-of-freedom velocity field is then applied to the headpart's center of gravity to drive the anatomic brain model, using the assumption of a rigid skull, to replicate the dummy headpart's kinematic response and to approximate the dynamic loading experienced in the actual crash test. The resulting soft tissue strains are evaluated as described in [2].

This approach provides a basic methodology for characterizing the dynamic loads sustained by the head, as measured with the NAP in a relatively rigid dummy headpart, and applying these loads to an analytical model of the brain. In this manner, soft tissue strains can be computed in response to dynamic loads measured in the actual crash test environment. *The generalized six-degree-of-freedom kinematic loading extracted from the NAP data reflects all dummy headpart impacts with the surrounding structure during the impact event, as well as dummy/neck reaction forces.* The compliance of the dummy headpart's skin is also reflected in the kinematic response.

The methodology presented is applicable to pure inertial loadings sustained in non-contact events, as well as for generalized head contact with interior surfaces and/or airbag impacts. Supplemental data characterizing angular and translational velocities and accelerations in body and in fixed coordinates is also generated during the transformation process. Kinematic quantities transformed to the fixed coordinate system may be particularly useful for comparing relative kinematics from one or more tests. The resulting data may also be used in conjunction with a headpart model to graphically illustrate and compare headpart kinematic response from separate tests, to illustrate headpart kinematics obscured by airbag deployment, and/or plot time history variables describing headpart kinematic response (e.g., angular position) versus time.

It is important to note that since impact forces acting on the skull are not directly indicated by the NAP data (although they may be implied), the above approach is not directly applicable to predicting detailed skull loading or fracture. The latter consideration, which requires a detailed description of localized loading and skull structural and material properties, is currently approximated by HIC. The methodology described herein is aimed at replicating kinematic response data which may be used in

conjunction with an anatomic brain model to estimate strains developed in soft brain tissue in response to dynamic loads sustained in real world collisions.

In [4], various accelerometer configurations were reviewed and the 3-2-2-2 configuration was determined to be the most reliable method for characterizing the generalized six-degree-of-freedom headpart kinematics, and equations were developed which can be used to effectively compensate for cross-products of angular velocity and centripetal accelerations introduced by the non-coplanar geometry of the 3-2-2-2 NAP configuration. These equations have been integrated into the coordinate transformation process to compensate for effects of non-coplanar geometry.

2. OVERVIEW

The following sections describe the major assumptions and operating principles used in the coordinate transformation process in chronological order of development. Initially, the coordinate transformation process and computer program described below was developed assuming that accelerometers within the 3-2-2-2 array were ideally located such that all transducers were located on axes defined by three orthogonal intersecting planes passing through the headpart center of gravity. This is referred to as the *coplanar* geometry configuration, and does not account for the small variations from actual accelerometer locations in the *non-coplanar* nine accelerometer array. Initial comparisons of simulated kinematics made using the corresponding coplanar geometry assumption, although qualitative in nature, compared very favorably with crash test film data and was certainly representative of the kinematics involved.

In order to study the effects of non-coplanar geometry, simulated accelerometer data was developed based on rigid body kinematics, to compute accelerometer data at each transducer location using both coplanar and non-coplanar geometries. Key to this approach was the use of kinematic scenarios with a known final position and angular orientation, which could be used to study variations in kinematics and accumulated strains developed in the anatomic model for coplanar versus non-coplanar geometry assumptions. Effects of typical transducer measurement errors were similarly evaluated. This lead to the development of a compensation algorithm, which was subsequently coded into the coordinate transformation software package, to essentially eliminate the effects of cross-products of angular velocity and centripetal accelerations associated with the actual non-coplanar geometry of the 3-2-2-2 NAP configuration.

A discussion of the principles used in the coordinate transformation process, with the initial assumption of ideal or coplanar geometry, is described below. The implementation of these principles is then described in terms of FORTRAN computer program variables followed by a profile of the resulting translational and rotational kinematic response quantities computed in body coordinates and/or transformed to fixed coordinates.

Regardless of the geometry assumption, the principal objective is to develop a method of tracking the relative angular orientation of the rotating, decelerating headpart with respect to the fixed coordinate system through time. This is accomplished by computing an angular velocity rotation matrix which describes instantaneous angular velocity components of the body coordinate system, and which allows the computation of an updated direction cosine matrix at each instant in time. This discussion is followed by recommendations for screening and preparing accelerometer data for the transformation process, required program inputs, and a typical interactive session.

Using actual crash test data, Section 3 describes the general procedure for (a) extracting headpart kinematic response in the form of translational and rotational velocity time histories at the headpart center of gravity, and (b) applying generalized translational and rotational velocities to an anatomic model to replicate the test kinematics, and consequently, the dynamic loads associated with the head impact. Comparison of simulated kinematics with available film data is provided from several views for selected crash tests to provide a general indication of actual versus simulated kinematics. As noted above, these initial simulations were made using the ideal or coplanar geometry assumption, and results compared very favorably with crash test film data.

To provide a more rigorous comparison of the transformation process, simulated NAP data is developed as described in Section 4, to rotate a rigid object about the fixed lateral axis from an initial orientation into a known final position. Rigid body angular kinematics are prescribed about a single fixed coordinate axis, with the headpart initially rotated about each body axis to produce simultaneous angular kinematics about each axis in the body coordinate system. The simulated accelerometer data is then processed using the transformation procedure and applied to the center of gravity of a finite element representation of a rigid body. The final position of the rigid object driven by the transformed accelerometer data is compared with the known final position of the rigid object to verify the transformation process.

Finally, an algorithm which compensates for the non-coplanar location of accelerometers in the NAP array is described. This considers effects of centripetal accelerations as well as Coriolis accelerations, or cross products of angular velocity, arising from the non-coplanar geometry of the NAP. This approach is useful to assess the effects of non-coplanar geometry, as well as transducer errors such as cross-axis sensitivity, on the final position of the transformed object, and to assess the relative effects of various error sources on angular kinematics.

2.1 PRINCIPLES OF THE TRANSFORMATION PROCESS

The transformation of quantities between coordinate systems is usually a simple task. Transforming components of a vector $\vec{R}$ from a non-rotating body coordinate system to a fixed coordinate system is done simply using:

$$R_{fi} = T_{ij} R_{bj} \qquad 2.1\text{-}1$$

where

$R_{fi} = i^{th}$ component of the vector $\vec{R}$ in the fixed coordinate system

$R_{bj} = j^{th}$ component of the vector $\vec{R}$ in the body coordinate system, and

$\left[T_{ij}\right] = \underset{\sim}{T}$ direction cosine matrix relating vector components from body-to-fixed coordinate system

However, when one coordinate system is rotating, as in the case of a dummy headpart, the rate of change of the rotating coordinate system must also be considered in the transformation of vector components between fixed and rotating coordinate systems. To distinguish matrices, vectors, and their components in the following discussion, vector quantities are indicated with the superscript ($\bar{}$), matrices by the subscript ($\underset{\sim}{}$), and vector or matrix components by subscripted indices. Repeated indices indicate summation. Also, reference to axes of the body coordinate system are generally indicated by lowercase subscripts *x, y,* and *z* while fixed coordinate system axes are generally indicated by uppercase *X, Y,* and *Z.*

Components of the time-varying vector $\dot{\bar{R}}_b$ when viewed or transformed to inertial coordinates are:

$$\dot{R}_{fi} = T_{ij}\dot{R}_{bj} + \dot{T}_{ij}R_{bj} \qquad 2.1\text{-}2$$

The first term on the right-hand side represents the transformation of time-varying components of the vector $\dot{\bar{R}}_b$ as if the body coordinate system was not rotating. The second term represents changes in the relative angular position of the body coordinate system, as if the there was no change in the vector $\dot{\bar{R}}_b$.

Hence, the appropriate expression for transforming a time-varying vector from the rotating body coordinates to a fixed, inertial coordinate system is given by equation 2.1-2, which is the matrix form of the Coriolis Equation, expressed in vector notation as:

$$\dot{\bar{R}}_f = \dot{\bar{R}}_b + \vec{\omega} \times \vec{R}_b \qquad 2.1\text{-}3$$

where

$\dot{\vec{R}}_f$ and $\dot{\vec{R}}_b$ are time-varying vector quantities measured in fixed and body coordinates, respectively, and

$\vec{\omega}_b \times \vec{R}_b$ is a supplemental vector required because the body coordinate system is in rotation.

The components of the cross product $\vec{\omega} \times \vec{R}_b$ in the body coordinate system is the vector quantity

$$(\vec{\omega} \times \vec{R}_b) = \omega_{jk} R_{kb} \qquad 2.1\text{-}4$$

where ω_{jk} are the elements of the rotation matrix $\underset{\sim}{\omega}$

$$\underset{\sim}{\omega} = [\omega_{jk}] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \qquad 2.1\text{-}5$$

The elements ω_x, ω_y, and ω_z are the instantaneous components of the angular velocity vector in the body coordinate system. It may be noted that the null values on the diagonal of the matrix $\underset{\sim}{\omega}$ result from the fact that angular velocities about each principal body axis do not affect angular velocity components about the corresponding principal axis in the fixed coordinate system. The sense of each term in $\underset{\sim}{\omega}$ is determined from considering directional effects of angular rotations in a right-handed coordinate system.

Transforming components of the vectors $\dot{\vec{R}}_b + \vec{\omega} \times \vec{R}_b$ to fixed coordinates, we obtain

$$\dot{R}_{fi} = T_{ij}(\dot{R}_{bj} + \omega_{jk} R_{bj}) \qquad 2.1\text{-}6$$

Comparing this with equation 2.1-2, we note $\dot{T}_{ij} = T_{im}\omega_{bmj}$ where *m* is a dummy index used in multiplication. This may also be expressed in matrix form as:

$$\underset{\sim}{\dot{T}} = \underset{\sim}{T}\,\underset{\sim}{\omega} \qquad 2.1\text{-}7$$

Equation 2.1-7 provides an algorithm which may be used to compute and update the orientation of the body coordinate system relative to fixed coordinates based on incremental changes in body angular velocity. For small incremental angular rotations at each time step, equation 2.1-7 can be integrated as indicated on the next page:

$$(\underset{\sim}{T}_{new} - \underset{\sim}{T}_{old})/\delta t = (\underset{\sim}{T}_{old} \cdot \underset{\sim}{\omega}) \qquad 2.1\text{-}8$$

or,

$$\underset{\sim}{T}_{new} = \underset{\sim}{T}_{old} + (\underset{\sim}{T}_{old} \cdot \underset{\sim}{\omega})\delta t \qquad 2.1\text{-}9$$

Having updated the new body coordinate system location, the transformation from body to fixed coordinates can proceed using the simple transformation equation in 2.1-1.

Using this marching algorithm, the new position of the body coordinate system is updated at each time step relative to fixed coordinates, in terms of its previous location, and the current angular velocity rotation matrix $\underset{\sim}{\omega}$, and the updated T matrix $(\underset{\sim}{T})_{new}$ is used to transform between rotating body and fixed coordinate systems. An excellent discussion of the mathematical treatment of this subject is provided by Broxmeyer in [5].

To transform time-varying vector quantities from body-to-fixed coordinates, update the position of the coordinate system at each time step and minimize error buildup, changes in angular orientation of the body coordinate system at each time step must be limited. In this implementation, the vector sum of body angular velocities at each time step is limited to one milliradian/sec (0.06 degrees/sec) per time step.

$$|\omega_b| = \sqrt{\omega_{b_x}^2 + \omega_{b_y}^2 + \omega_{b_z}^2} \leq 0.001 \text{ radians/sec} \qquad 2.1\text{-}10$$

If the one milliradian limit is exceeded, the time interval is subdivided to limit the maximum change in body angular velocity at each time step. The implementation of this process is described below.

2.1.1 Implementation and Program Variables

The translational body accelerations are read into the internal arrays *bacc1* through *bacc9* in the main program. To facilitate the computation of angular kinematics, accelerometers were initially assumed to be perfectly located at the ideal coplanar locations shown in Figure 1-2, with sense axes of multiple accelerometers co-located at a single point in space. Because of the actual non-coplanar geometry of transducer array, this assumption could introduce some spurious acceleration components into the calculation because centripetal accelerations and cross-products of angular velocities are not computed exactly. These variations are discussed in Section 4 which also presents an algorithm to compensate for these spurious effects.

The above accelerations are used with the ideal geometry approximation to estimate body angular accelerations *bomegdi* ($i = x,y,z$), about the body x, y, and z axes, respectively.

This is done by averaging pairs of accelerometers in the 3-2-2-2 array as illustrated in Figure 1-2, and as shown below.

$$\text{bomegdx} = 0.5*[((\text{bacc4} - \text{bacc3})/\text{ry} + (\text{bacc2} - \text{bacc9})/\text{rz})]*G$$

$$\text{bomegdy} = 0.5*[((\text{bacc3} - \text{bacc6})/\text{rx} + (\text{bacc8} - \text{bacc1})/\text{rz})]*G \qquad 2.1\text{-}11$$

$$\text{bomegdz} = 0.5*[((\text{bacc7} - \text{bacc2})/\text{rx} + (\text{bacc1} - \text{bacc5})/\text{ry})]*G$$

where G = gravitational acceleration constant, and
rx, ry, and rz are distances from the cg to accelerometers located on x, y, and z arms respectively.

Body angular velocities are then computed by integrating body angular accelerations, and treating the body coordinate system as instantaneously fixed in space for an infinitesimal incremental rotation. This approximation is made by limiting incremental changes in the vector sum of body angular velocity components Bomeg*i* *(i= x,y,z),* to one milliradian (i.e., 0.06 degrees) /sec or less, for each point in the input accelerometer data.

$$\text{Bomeg}i_{new} = \text{Bomeg}i_{old} + \text{Bomegd}i \times \delta t \qquad i = x, y, z \qquad 2.1\text{-}12$$

$$\text{Bomegv} = |\omega_b| = \sqrt{\text{Bomegx}^2 + \text{Bomegy}^2 + \text{Bomez}^2} \le 0.001 \text{ radian} \qquad 2.1\text{-}13$$

Since the data sampling rate is typically 8 kHz or more, incremental rotations typically do not exceed one milliradian/sec. However, in the event this should occur, the data is linearly interpolated and subdivided into smaller increments to impose the one milliradian/sec increment limit.

It is assumed that the body and fixed coordinate systems are either initially co-incident at the body center-of-gravity or are related by the sequential rotations *phi, theta,* and *psi* about body x, y, and z axes, respectively. In either case, the initial angular relationship between body and fixed coordinate systems is indicated by the direction cosine matrix of 2.1-14.

$$\begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \qquad 2.1\text{-}14$$

Since an initial angular displacement of the body coordinate system is assumed relative to fixed coordinates, the inverse or transform of the above matrix is required to relate body position relative to fixed coordinates.

$$\underset{\sim}{T}_f^b = \left[\underset{\sim}{a}_f^b\right]^{-1} = \underset{\sim}{a}_b^f$$

or

$$\underset{\sim}{T}_{ij} = \underset{\sim}{a}^{-1}{}_{ij} = \underset{\sim}{a}_{ji} \qquad 2.1\text{-}15$$

For null values of *phi, theta,* and *psi,* T and a reduce to the identity matrix.

The instantaneous body angular velocities and the current position of the body coordinate system, described by direction cosines, are used to compute the incremental change in angular orientation of the body coordinate system for each time interval according to:

$$\underset{\sim}{TDT}_{ij} = \sum_{k=1}^{3} T(i,k) \times Bomeg(k,j) \qquad 2.1\text{-}16$$

where

Bomeg(k,j) = angular velocity along the *k*th body axis relative to *j*th fixed axis direction

The angular relationship between body and fixed coordinate systems is then "updated" by rotating the body coordinate system according to the incremental angular displacement occurring during that interval. The updated direction cosine matrix relating the angular position of the body coordinate relative to the fixed axis system is computed by

$$\underset{\sim}{T}_{i,j_{new}} = \underset{\sim}{T}_{i,j_{old}} + \underset{\sim}{TDT}_{i,j} \times \delta t \qquad 2.1\text{-}17$$

where δt is the time increment. The new or updated position of the body coordinate system is taken as "temporarily fixed" for the next increment of computed angular velocities, and the updated coordinate location and new (infinitesimal) angular velocities are used to compute the next change in angular orientation of the body coordinate system. This process is continuously repeated.

At the completion of the process, direction cosines relating the angular orientation of principal body and fixed axes are written to the output files

F0001DCOS.011 F0001DCOS.022 F0001DCOS.033

and the angular orientation of rigid body axes relative to fixed coordinates is established by simultaneous solution of equation 2.1-14 for the angles *phi, theta,* and *psi*. The time histories of these angular displacements *phi, theta,* and *psi* are respectively written to the output files:

F0001ROTD.00X F0001ROTD.00Y F0001ROTD.00Z

2.1.2 Transformation from Body to Fixed Coordinates

After the angular relationship between body and fixed coordinates has been established using the above procedure, it is a simple matter to transform kinematic quantities computed in the body coordinate system to fixed coordinates. For example, translational accelerations at the center of gravity with respect to fixed coordinates are computed according to:

$$\begin{aligned} \text{Facc}X &= \sum_{j=1}^{j=3} \text{T}(1,j) \times \text{Bacc}j \\ \text{Facc}Y &= \sum_{j=1}^{j=3} \text{T}(2,j) \times \text{Bacc}j \\ \text{Facc}Z &= \sum_{j=1}^{j=3} \text{T}(3,j) \times \text{Bacc}j \end{aligned} \qquad \text{2.1-18}$$

where Bacc*j* = accelerations along body *x, y,* and *z* axes, and
Facc*I* = acceleration along fixed *X, Y,* and *Z* axes, etc.

Translational accelerations with respect to fixed axes are written to the output files:

F0001AM00.00X F0001AM00.00Y F0001AM00.00Z

Once transformed to fixed coordinates, translational accelerations at the CG are integrated to provide translational velocities at the CG with respect to fixed coordinates. Translational velocities in fixed coordinates are stored in the arrays Fvel*X,* Fvel*Y,* and Fvel*Z* and these quantities are written to the output files:

F0001VM00.00X F0001VM00.00Y F0001VM00.00Z

Translational velocities along the fixed X, Y, and Z axes are also written in *time-value* pairs to the output file LC_FILE.DAT, in a format compatible with specifying rigid body motion in the explicit finite element code LS-DYNA3D. For compatibility with the anatomic model, English units of in/sec are used.

Angular accelerations and velocities computed in the body coordinate system are also transformed to fixed coordinates in the same manner. Angular accelerations about the fixed X, Y, and Z axes are computed by:

$$\begin{aligned} \text{Fomegd}X &= \sum_{j=1}^{j=3} \text{T}(1,j) \times \text{Bomegd}j \\ \text{Fomegd}Y &= \sum_{j=1}^{j=3} \text{T}(2,j) \times \text{Bomegd}j \\ \text{Fomegd}Z &= \sum_{j=1}^{j=3} \text{T}(3,j) \times \text{Bomegd}j \end{aligned} \qquad \text{2.1-19}$$

These quantities are written to the output files:

F0001ROTA.00X F0001ROTA.00Y F0001ROTA.00Z

Angular velocities in fixed coordinates are obtained from:

$$\begin{aligned} \text{Fomeg}X &= \sum_{j=1}^{j=3} \text{T}(1,j) \times \text{Bomeg}j \\ \text{Fomeg}Y &= \sum_{j=1}^{j=3} \text{T}(2,j) \times \text{Bomeg}j \\ \text{Fomeg}Z &= \sum_{j=1}^{j=3} \text{T}(3,j) \times \text{Bomeg}j \end{aligned} \qquad \text{2.1-20}$$

and are stored in the output files:

B0001ROTV.00X B0001ROTV.00Y B0001ROTV.00Z

and are also written in *time-value* pairs to the output file LC_FILE.DAT with units of rad/sec.

The output file LC_FILE.DAT now contains a description of the generalized six-degree-of-freedom motion at the center of gravity of a dummy headpart, in terms of translational and angular velocities. This file was generated to be included with the input file of the anatomic brain model described in [3] and is used to specify the generalized six-degree-of-freedom kinematics and replicate the dynamic loads experienced by the soft tissue of the brain in response to head impact collision loading measured in the form of NAP data.

2.2 DATA PREPARATION AND INPUT

The data should initially be checked for bias error and for overall data quality, and to assure that all nine accelerometer data channels are available. Data acquisition errors such as spurious spikes in the data, saturation of the measurement signal, scale factor errors and/or polarity errors will obviously degrade the usefulness of the data. To the extent possible, bias errors should be assessed and corrected prior to utilizing the data. Overplotting the three accelerations along the body x (bacc1, 5 and 8), y (bacc2, 6 and 9), and z (bacc3, 4 and 7) will often reveal anomalies in the data since these accelerations are generally very similar in character.

Also, since most acceleration time history data includes a "dead-time" or zero-acceleration state corresponding pre-impact run-in time, it may be desirable to time-shift the data so as to separate pre- and post-impact domains. This will minimize computation time and focus results on the interval corresponding to head impact. This is particularly important if the resulting translational and rotational velocities at the CG are to be used to drive an anatomic model, since this can be computationally intensive. However, this step may not be very important if the objective is to simply generate kinematic response data or to replicate the kinematics by driving a rigid headpart without the deformable interior components.

Required Program Inputs

2.2.1 Initial Headpart Orientation Relative to Inertial Axes

To replicate dynamic loads, only the NAP data is required regardless of headpart orientation. However, for a rotated headpart, replicating both dynamic loads and the specific kinematic response as observed from a particular inertial location requires (a) relating the rotated headpart body coordinate system to the fixed coordinate system, and (b) applying the dynamic loads to the anatomic model placed in the identical angular orientation. Although this process is not required to replicate dynamic loads, this may be useful for the comparison of kinematic responses in a particular reference frame.

Since NAP sense axes are typically referenced to the vehicle coordinate system, the nominal orientation of the headpart coordinate system is assumed to be that of the vehicle coordinate system, and the initial angular orientation of body and fixed coordinate systems are coincident.

This assumption indicates that accelerometers along the body z (yaw) axis are nominally perpendicular to the earth's gravitational field, while the body x (roll) axis is aligned with the vehicle longitudinal axis, and the body y (pitch) axis is aligned with the vehicle lat-eral axis. Probably the most common deviation from this initial angular orientation is for the headpart to be rotated about its pitch or lateral axis relative to the vehicle longitudinal axis.

The direction cosine matrix (Eq. 2.1-14) was included in the transformation procedure to provide a more general initial angular relationship between body and fixed coordinate systems. The program first prompts the user for a set of angles describing the initial angular position of the headpart relative to fixed coordinates, in terms of sequential rotations about the body x (lateral), y (pitch), and z (yaw) axes. If the dummy headpart is aligned with the vehicle x, y, and z axes, (or if the objective is primarily to replicate dynamic head loads), null values should be entered. If the headpart is rotated about only one axis, the angular rotation should be entered (in radians) for the corresponding rotation. For example, if the headpart were rotated 0.1 radians about the body y or pitch axis, the following values would be input:

PHI, THETA, PSI (rad): angle of rotation of accelerometer
system about, (1) x-axis, (2) y-axis, and (3) z-axis
...PHI, THETA, PSI

> 0.0, 0.10, 0.0

A rotation of 0.5 radians or 28 degrees about the z or vertical (yaw) axis, which might be representative of a driver-side dummy headpart rotated to face an A-pillar, would be entered as:

PHI, THETA, PSI (rad): angle of rotation of accelerometer
system about, (1) x-axis, (2) y-axis, and (3) z-axis
...PHI, THETA, PSI

> 0.0, 0.0, 0.5

Normally, the headpart is either aligned with the vehicle axes or rotated about only one axis.

2.2.2 Accelerometer Data Input

The transformation software is currently coded to read and write data files formatted according to the NHTSA User Data Set (UDS) file format. The software must be modified to read other data formats, and comment cards have been included in the program listing to facilitate these modifications. The NAP data must also be read in the specific sequence indicated in Table 2-1 or keyed to positional information, if available, contained within accelerometer data file.

Acceleration units should be in G's and output files generated have English units. Translational velocities are in inches/sec and angular velocities are in rad/sec, to be consistent with English units used in the anatomic brain model.

TABLE 2-1. SEQUENCE FOR READING NAP DATA BY ARRAY LOCATION

Input File No.	Location (CG or Arm)	Sense Axis
1	CG	X
2	CG	Y
3	CG	Z
4	Y Arm	Z
5	Y Arm	X
6	X Arm	Z
7	X Arm	Y
8	Z Arm	X
9	Z Arm	Y

2.2.3 Typical Interactive Session

A complete interactive session is illustrated below. The angles PHI, THETA, and PSI represent sequential rotations about the body x, y, and z axes respectively, relating the initial orientation of the dummy headpart to fixed coordinates. In this session, the headpart is rotated 0.1 radians about the lateral or pitch axis, and the acceleration channel number is distinguished by the last character in the input file string. A sequence of input files with the base filename V0001AM00.00*i* (*i=1-9*) represents the nine accelerometer package data. Assuming accelerometer locations within the NAP array as defined in Figure 1-2, accelerometer data would be input in the sequence illustrated below. The command line for execution in the VAX environment is: run LC_COMP.

*** Transform NAP Data From Body to Inertial Coordinates ***

-- PHI, THETA, PSI (rad): angle of rotation of accelerometer
system about, (1) x-axis, (2) y-axis, and (3) z-axis
...PHI/(E,F), THETA/(E,F), PSI/(E,F):

0.0,0.1,0.0

Enter 1st UDS file in NAP array (Xcg)
Control-Z to Stop >> V0001AM00.001

Enter 2nd UDS file in NAP array (Ycg)
Control-Z to Stop >> V0001AM00.002

Enter 3rd UDS file in NAP array (Zcg)
Control-Z to Stop >> V0001AM00.003

Enter 4th UDS file in NAP array (Yarm Zg)
Control-Z to Stop >> V0001AM00.004

Enter 5th UDS file in NAP array (Yarm Xg)
Control-Z to Stop >> V0001AM00.005

Enter 6th UDS file in NAP array (Xarm Zg)
Control-Z to Stop >> V0001AM00.006

Enter 7th UDS file in NAP array (Xarm Yg)
Control-Z to Stop >> V0001AM00.007

Enter 8th UDS file in NAP array (Zarm Xg)
Control-Z to Stop >> V0001AM00.008

Enter 9th UDS file in NAP array (Zarm Yg)
Control-Z to Stop >> V0001AM00.009

At the end of the computation, a list of files is printed that describes supplemental kinematic variables with respect to fixed or body coordinates. Translational and angular velocity time histories at the headpart CG are also written to the separate file, LC_FILE.DAT in the form of *Load Curves* to specify prescribed headpart kinematics in finite element simulations. The following message is printed when computations are completed.

The Following Output Files Have Been Created:

Translational Accelerations - Inertial Coordinates.
F0001AM00.00X F0001AM00.00Y F0001AM00.00Z

Rotational Velocities - Body Coordinates
B0001ROTV.00X B0001ROTV.00Y B0001ROTV.00Z

Rotational Accelerations - Inertial Coordinates
F0001ROTA.00X F0001ROTA.00Y F0001ROTA.00Z

Direction Cosines Relating Principal Axes
F0001DCOS.011 F0001DCOS.022 F0001DCOS.033

Rotational Displacements - Inertial Coordinates

F0001ROTD.00X	F0001ROTD.00Y	F0001ROTD.00Z

Rotational Velocities - Inertial Coordinates

F0001ROTV.00X	F0001ROTV.00Y	F0001ROTV.00Z

Translational Velocities - Inertial Coordinates

F0001VM00.00X	F0001VM00.00Y	F0001VM00.00Z

Load Curves Written to: LC_FILE.DAT

2.2.4 Supplemental Kinematic Outputs - Filenaming Convention

In the filenaming convention used above, those kinematic quantities beginning with the letter "F" are relative to inertial or fixed coordinates, and those beginning with the letter "B" are relative to the body coordinate system. The next four characters are generally reserved for a specific test number, and the middle four characters generally correspond to NHTSA conventions for describing kinematic variables, although many of the above kinematic quantities are unique to these calculations, as discussed further in Section 3. The last three characters form the suffix and generally indicate the direction or axis of the kinematic quantity. Filenaming conventions may easily be modified if desirable. Comment cards are included in the main program to indicate labeling of output filenames and kinematic quantities.

All output quantities are defined relative to the headpart center of gravity. Direction cosine files contain time histories describing the angular relationship between principal body and fixed coordinate axes, with the following suffix assignments:

F0001DCOS.011 indicating body x axis to fixed X axis
F0001DCOS.022 indicating body y axis to fixed Y axis
F0001DCOS.033 indicating body z axis to fixed Z axis

Typical kinematic output quantities resulting from the processing of NAP data are discussed in Section 3.

3. APPLICATION TO CRASH TEST DATA

To illustrate the process described above, the transformation software was applied to representative crash tests selected from the NHTSA crash test database. Tests were selected which had relatively familiar or predictable responses, good film data, and high quality NAP data. The NAP data was processed to transform dummy headpart accelerations to kinematic quantities computed at the headpart center of gravity, in the fixed coordinate system, and applied to a finite element model as described below.

3.1 GENERAL PROCEDURE FOR APPLYING HEADPART DYNAMIC LOADING

Although the following procedure is general, the anatomic model described in [3] is used to illustrate the application of dynamic loads in the form of specified translational and rotational kinematics at the headpart CG. Figure 3-1 is a cutaway view of the deformable internal components consisting of a visco-elastic representation of soft brain tissue and a stiffer, but deformable lining which also forms the falx anatomic partition. These components are contained within a rigid skull/headpart component. Subjecting the entire model to the test kinematics will result in the application of dynamic loads, in the form of inertial forces, to deformable interior components. The same procedure may be applied to the headpart only, to replicate the kinematics associated with contact or non-contact events as well. In simulations using this model, the pre-processor LS-INGRID [6] was used for mesh generation, the explicit finite element code LS-DYNA3D [7] was used for numerical integration, and LS-TAURUS [8] was used for post processing.

Dynamic loads are imposed on the deformable interior components of the model by specifying the kinematic motion of the rigid exterior skull/headpart containing the deformable components. This is done by applying generalized six-degree-of-freedom velocity boundary conditions at the rigid skull/headpart CG, constraining the rigid headpart to the same kinematics experienced in the test. Translational and rotational velocity time histories are computed at the CG location during the transformation process, in the form velocity boundary condition *load curves*, for convenient integration into the finite element code input file.

As noted above, the NAP data reflects all dummy headpart impacts with the surrounding structure during the impact event, as well as dummy/neck reaction forces. While the external surface of the headpart model is rigid, the compliance associated with the outer skin of the dummy headpart is not disregarded, since this is reflected in the NAP response. This approach is being developed because making direct and accurate measurements of soft tissue strains induced by rotational kinematics would be extremely difficult using a surrogate soft-tissue test device and measurement system.

CUTAWAY VIEW OF ANATOMIC BRAIN MODEL:

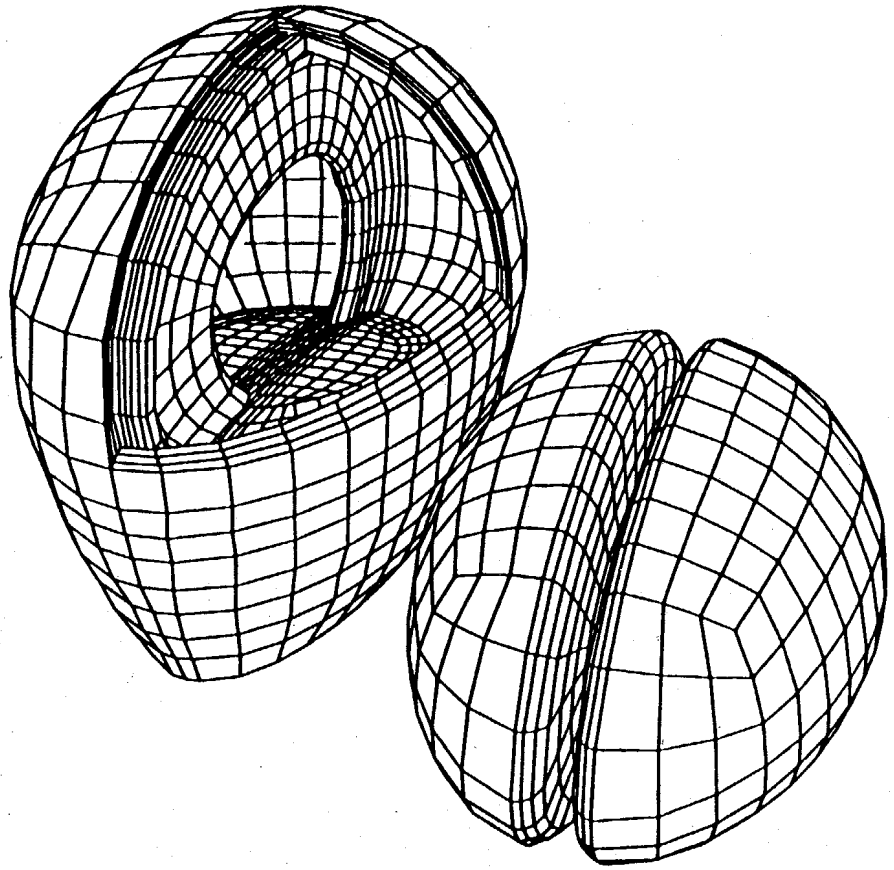


FIGURE 3-1. CUTAWAY VIEW OF ANATOMIC MODEL WITH DEFORMABLE INTERIOR COMPONENTS

Since impact forces imparted to the skull are not directly indicated by the NAP data (although they are implied), the above approach is not directly applicable to predicting skull fracture. The latter consideration, which requires a detailed description of localized loading and a much more complex skull model, is currently approximated by HIC. The methodology described herein is aimed at the computation strains developed in soft interior materials approximating brain tissue in response to dynamic loads sustained in real world collisions.

To include high frequency effects in imposing velocities at the model CG, each load curve is written using 1001 points. Since total simulation time is typically 100 msec, this typically results in a time interval very similar to the sample rate of the raw accelerometer data. The replication of high frequency content is apparent in the example which follows.

3.2 SELECTION OF CRASH TESTS FOR DEMONSTRATION

During the initial phase of this work, the NHTSA database was reviewed to define tests for which NAP data was available and several tests were considered to illustrate the process and compare kinematic responses extracted from the test data with film data describing headpart kinematics. The principal criteria for test selection included: completeness and quality (i.e., freedom from obvious measurement system error) of NAP data; predictable and familiar headpart kinematics; and quality of film data for assessing test versus simulated kinematics. Comparing simulated kinematics with film data can be very difficult because viewing angles and perspective are very difficult to duplicate.

In 1992, a review of the NHTSA database was made to identify NCAP, BASELINE, and COMPLIANCE tests with post-1985 test dates, which included NAP data. From these results, the most promising source for NAP data was the BASELINE test data group. Results from the database query were reviewed to assess factors such as collision mode, impact velocity, instrumentation, occupant seating, resulting HIC value, and restraint system. Film data was also requested for a subset of tests and reviewed for clarity. The following tests were selected for demonstration purposes.

3.3 OBLIQUE HEAD A-PILLAR IMPACT SIMULATION

Since there was a good base of test data available involving head impacts with upper interior surfaces such as the A-pillar (i.e., the forward post supporting the vehicle roof and securing the windshield on each side of the vehicle), a 1984 Ford Tempo sled test designed to produce head impact with the driver side A-pillar was the first demonstration case selected (NHTSA test number 1235). This impact scenario was quite familiar since head impacts with padded and unpadded A-pillars had been simulated [3] using another A-pillar design, and essential features of the kinematic response are fairly predictable due to the familiar geometry of A-pillars and their orientation.

From the test report documentation, the vehicle was a four door sedan, placed on the sled at an angle of 36 degrees clockwise, with respect to the sled axis. Figure 3-2 illustrates the relative positions of the sled buck on the sled track, side-mounted and wayside cameras, and the headpart in approximate relative position to the vehicle and cameras. An Alderson 50th percentile HYBRID III dummy was located at the left front driver position, and the object of the test was to simulate a 20 mph head impact with the vehicle's left A-pillar which was lined with production trim. This test was one in a series where head accelerations were compared to assess the effects of adding special foam padding materials on the A-pillar. The dummy was unrestrained.

3.4 DATA PREPARATION

To begin, the nine channels of accelerometer data were reviewed to check that all data channels were available and properly labeled. As a check for correct polarity and axis labeling, the three accelerations along each body axis were plotted as a group. The general character of the acceleration time histories along each axis are usually distinctive and appear similar. The data was also reviewed for bias and for data spikes and saturated signals. To remove null acceleration levels prior to impact, the data was then time shifted to the beginning of the acceleration pulse and truncated about 50 msec after the accelerations returned to zero to provide a longer time interval since angular kinematics usually lag peak accelerations. In terms of acceleration, the impact event was only about 20 msec long; however, to include post impact angular kinematics, a pulse duration of 100 msec was used and each channel was filtered at 500 Hz. Subsampling the data is not recommended, since this could affect accuracy.

Since the initial velocity of the sled and, hence, headpart is zero, the null initial velocity condition (included as a part of the accelerometer data file) was used. For barrier testing, the initial velocity condition is non-zero and must be input to the program if not available as supplemental data contained with the input file. (Since the initial velocity is contained

SLED BUCK ORIENTATION AND VIEWING ANGLES

CASE 2 - TEST 1235

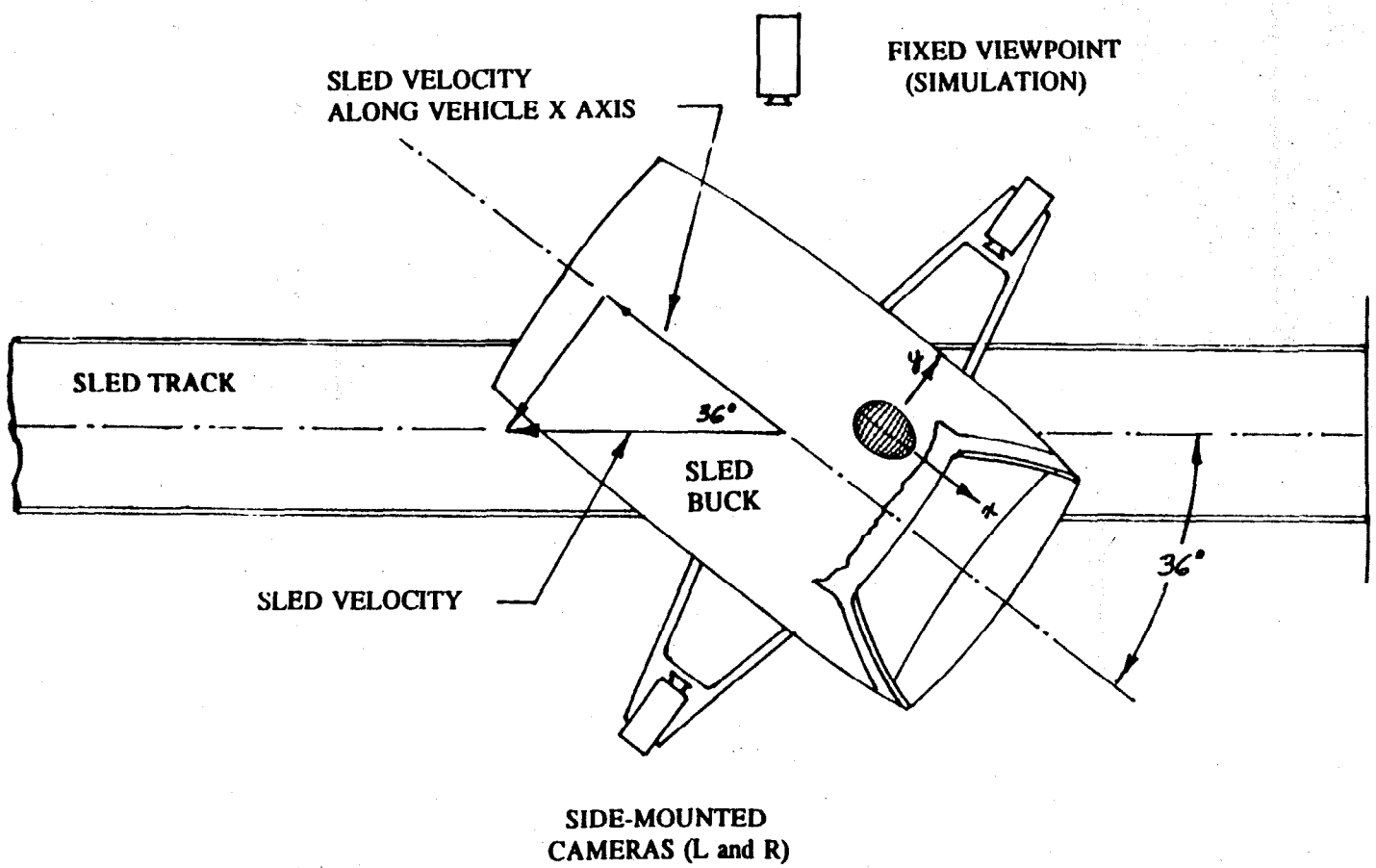


FIGURE 3-2. RELATIVE LOCATION OF CAMERAS AND SLED BUCK ON TEST TRACK

as a part of NHTSA UDS input files, the program does not prompt the user for this data.) The headpart orientation is taken to be initially coincident with vehicle axes and with no angular displacement between principal body and inertial axes. The location of the NAP array at the center of gravity of the rigid finite element headpart is shown in Figure 3-3.

3.5 OBLIQUE HEAD A-PILLAR IMPACT SIMULATION - RESULTS

In this collision scenario, the sled buck is rotated on its axis such that the A-pillar and the dummy headpart are approximately in line with the sled thrust axis. The sled buck is then accelerated rearward simulating an actual vehicle collision pulse, such that the A-pillar impacts the head of the dummy. Considering expected headpart kinematics, since a full dummy is being used, vertical displacement should be largely constrained by the lower dummy structure, and the motion of the headpart in the horizontal plane should be essentially along the thrust axis of the sled. As the dummy headpart strikes the A-pillar, the headpart is rotated to conform with inclined surfaces associated with and adjacent to the A-pillar. The resulting kinematics should reflect these impact mechanics.

Illustrations of simulated headpart kinematics with respect to inertial coordinates, overlayed in 20 msec intervals, are shown in Figures 3-4 through 3-7. Figures 3-4 and 3-5 are elevation views of headpart kinematics along the axis of impact (i.e., along the sled axis), from left and right sides respectively. Fully developed A-pillar contact occurs at about 40 msec and is very apparent from these views, with the head facing up to the A-pillar surface as it moves directly into the head. As expected, headpart kinematic response is primarily in the horizontal or lateral plane, due to constraints imposed by lower extremities of the dummy. Headpart angular rotations are also very similar to the angles of inclination typically seen in A-pillar impacts.

Figure 3-6 is a plan view of resulting headpart kinematics, indicating a head trajectory along the line of action of contact forces produced by the interaction of the sled-buck A-pillar component moving directly into the headpart. This produces a kinematic trajectory of 36 degrees with respect to the headpart longitudinal axis as implied by the test arrangement illustrated in Figure 3-2.

Figure 3-7 is a plan view of headpart angular kinematics in 12.5 msec intervals, with translational kinematics removed to isolate the rotational components of motion. In each tile, the headpart is centered in the post-processor viewing window, thus removing translational components of motion. These tiles are a sub-sample of stored kinematic states which can be played back in sequence to create a "movie" of the kinematic response. Figure 3-8 illustrates the corresponding angular kinematics experienced by the soft tissue component of the anatomic model within the rigid headpart.

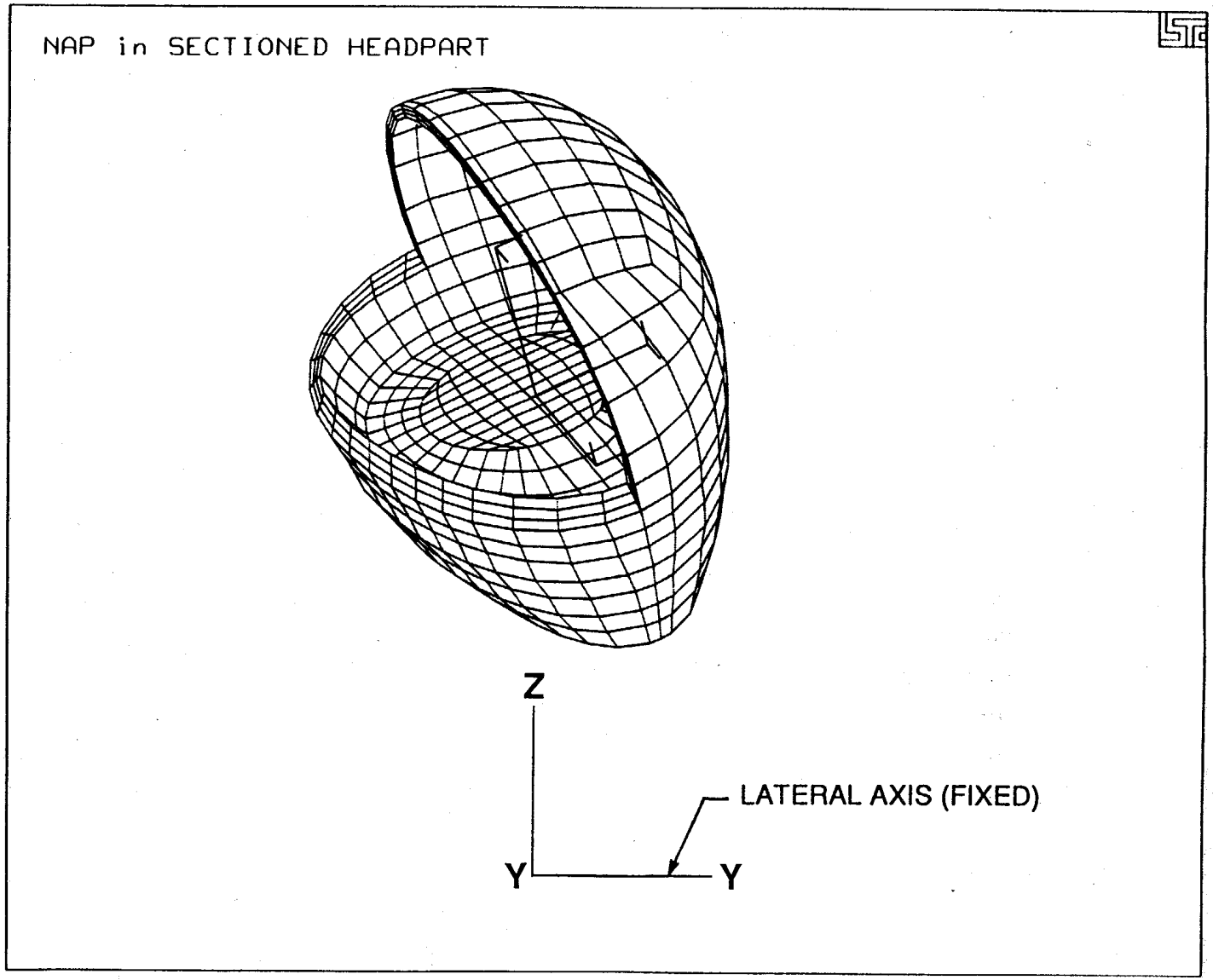


FIGURE 3-3. NAP ARRAY AT CG OF FINITE ELEMENT HEADPART

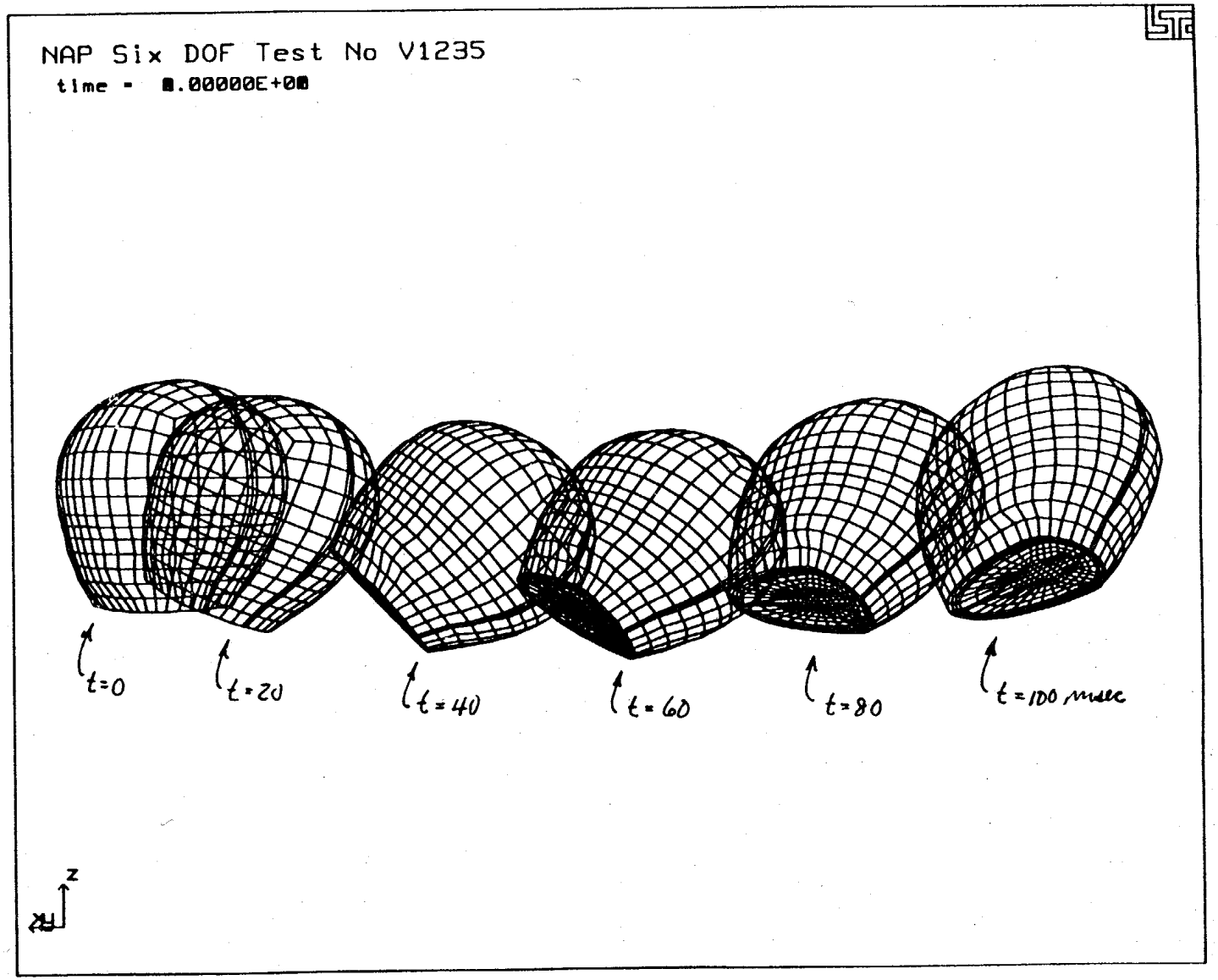


FIGURE 3-4. ELEVATION VIEW OF HEADPART KINEMATICS (LEFT SIDE) FOR OBLIQUE IMPACT TEST

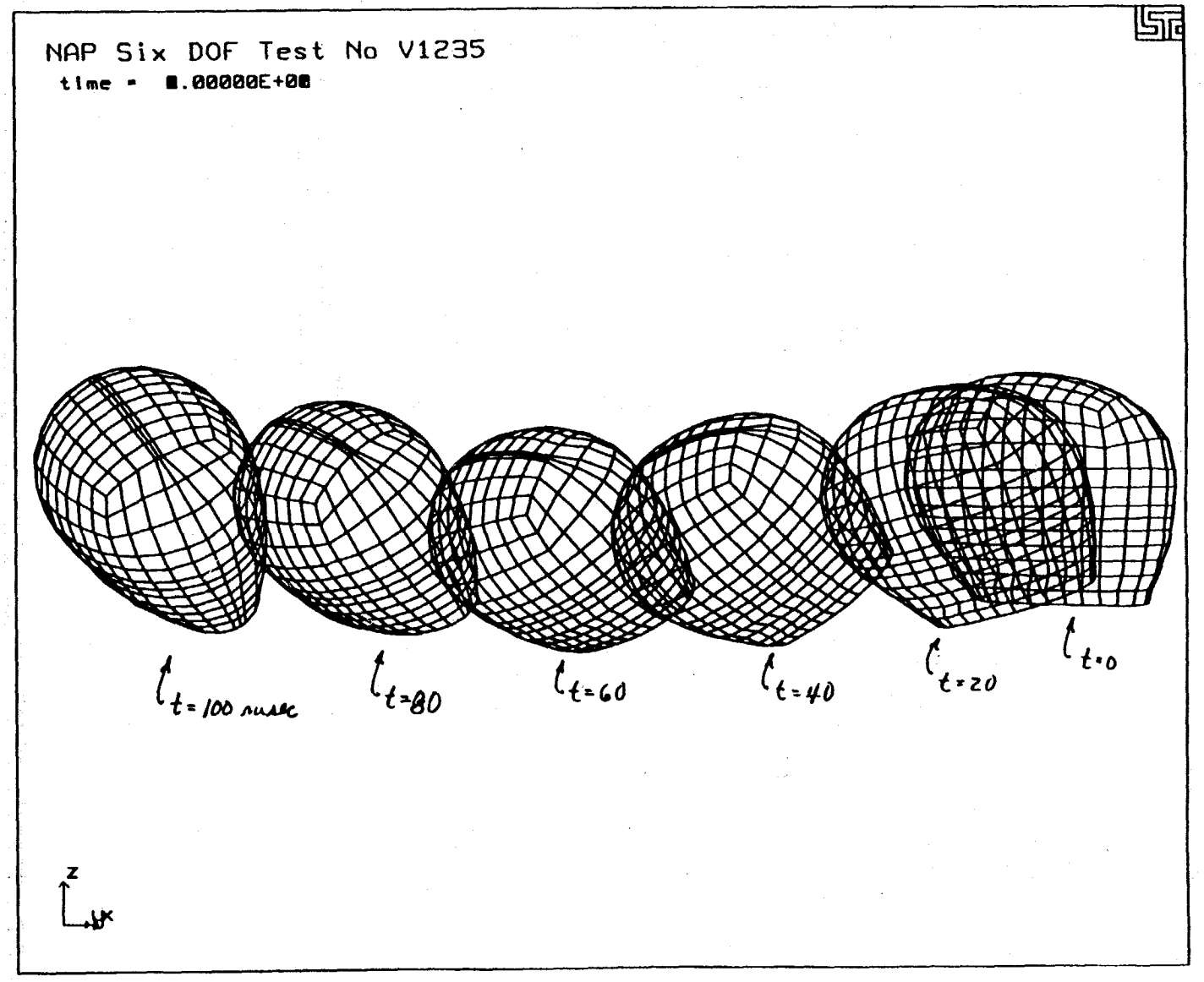


FIGURE 3-5. ELEVATION VIEW OF HEADPART KINEMATICS (RIGHT SIDE) FOR OBLIQUE IMPACT TEST

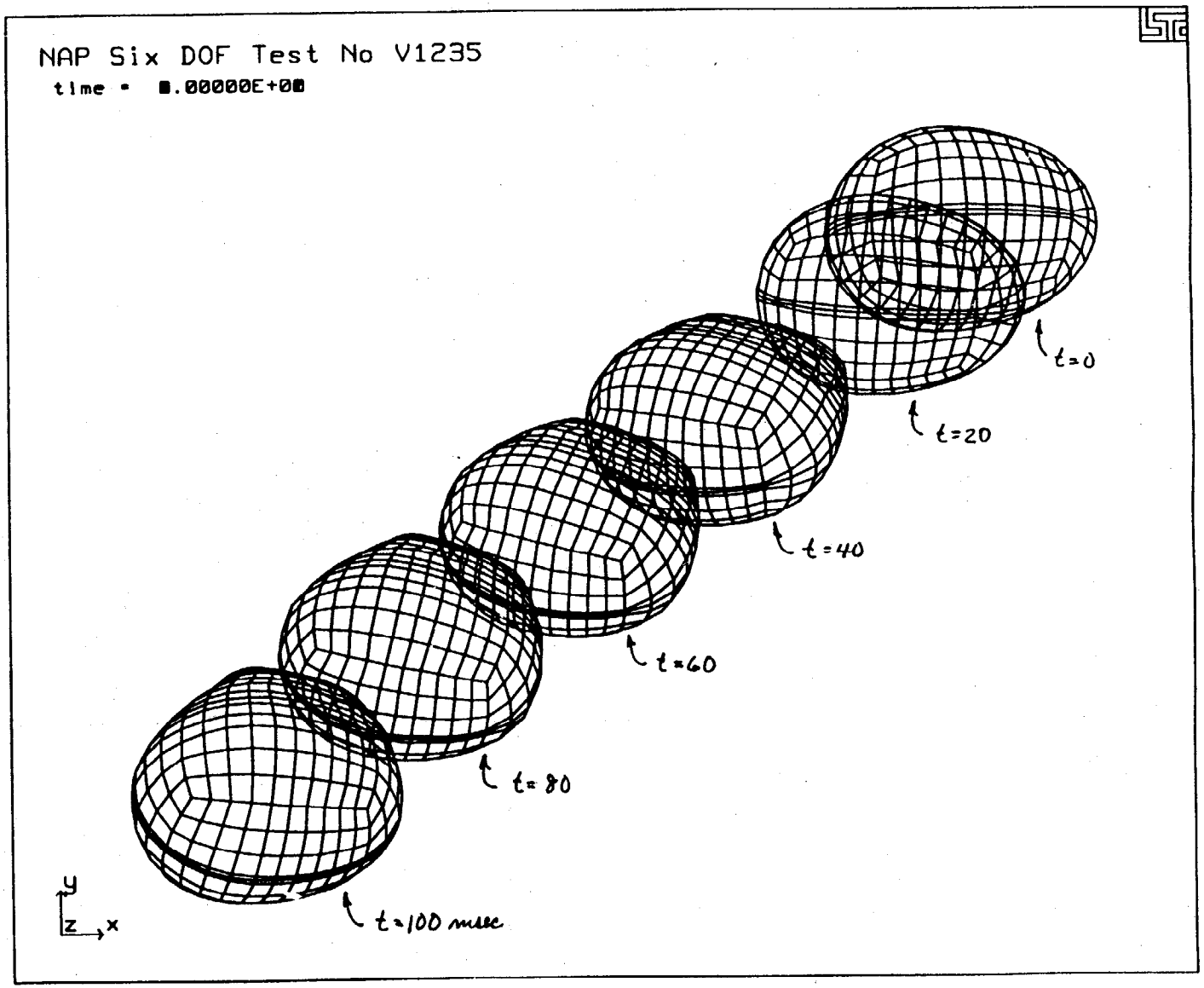


FIGURE 3-6. PLAN VIEW OF HEADPART KINEMATICS FOR OBLIQUE IMPACT TEST

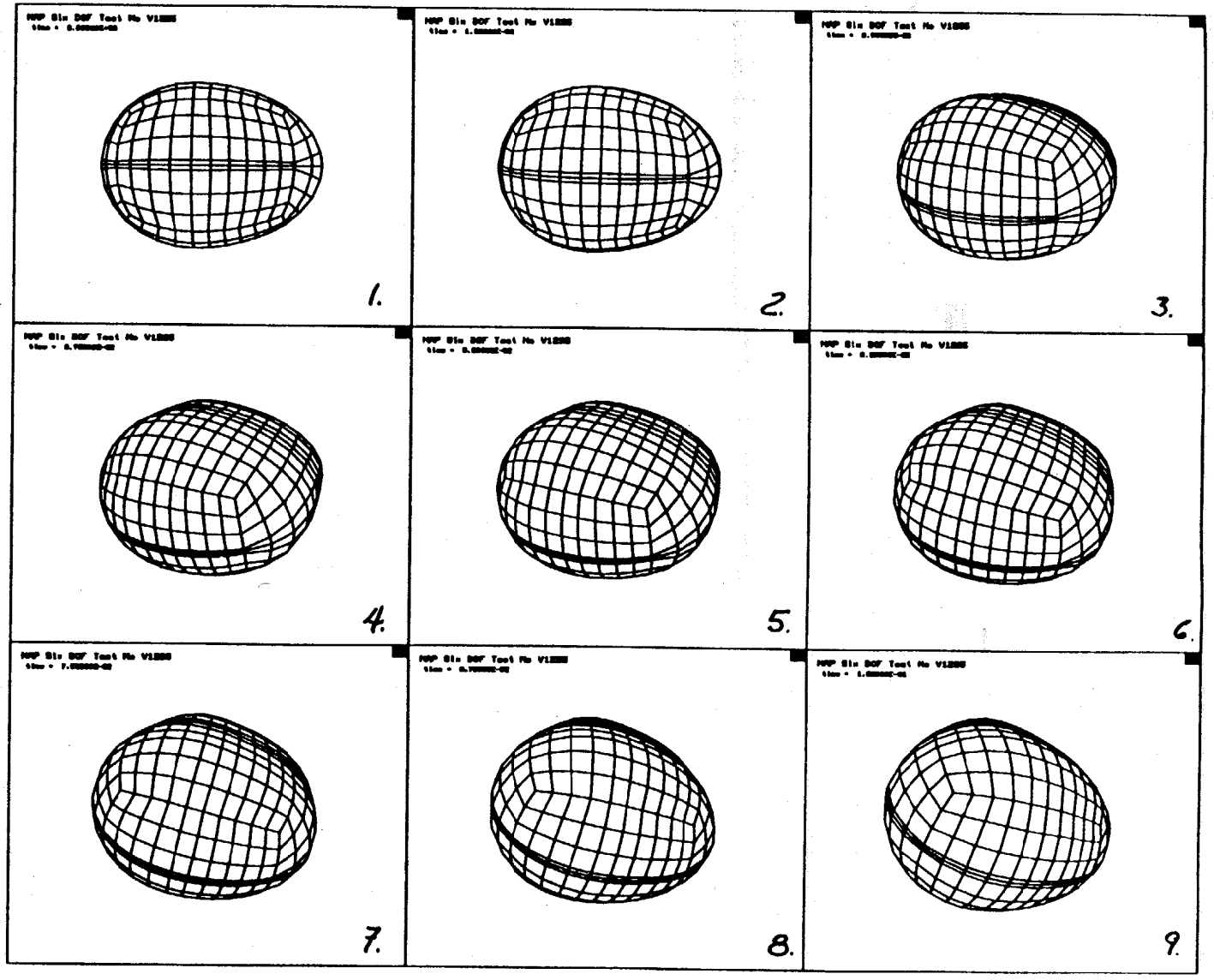


FIGURE 3-7. PLAN VIEW OF HEADPART KINEMATICS (ROTATIONS ONLY) FOR OBLIQUE IMPACT TEST

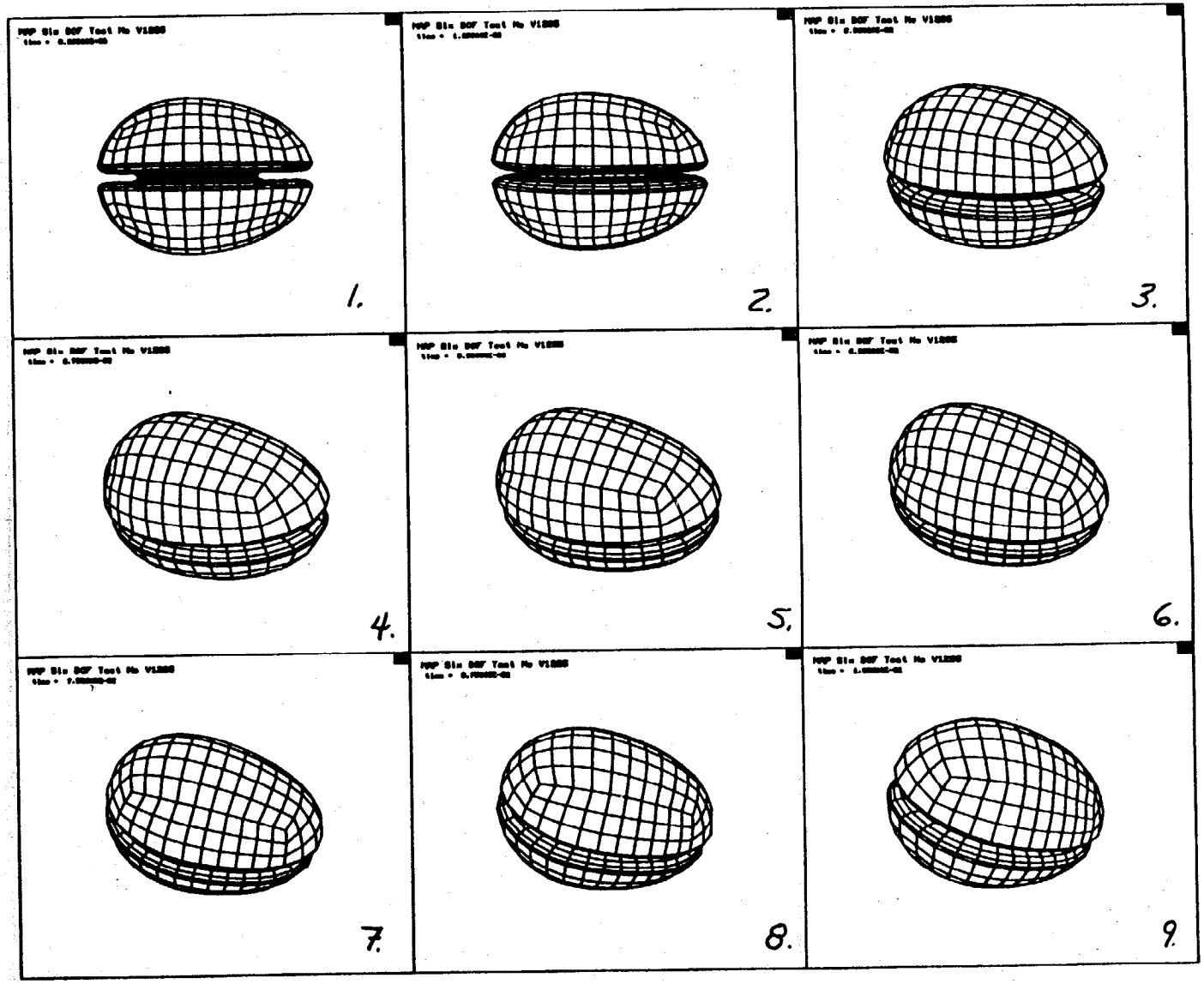


FIGURE 3-8. PLAN VIEW OF DEFORMABLE COMPONENT (ROTATIONS ONLY) FOR OBLIQUE IMPACT TEST

Using procedures developed by Bandak and Eppinger in [2], Figure 3-9 illustrates the accumulation of strain induced in the soft viscoelastic representation of brain tissue in terms of the volume fraction of material which has exceeded principal strain levels of 5, 7, 9, 11, and 13 percent. As discussed in [2], these strains are induced primarily by angular kinematics and these estimates are an initial step towards developing the capability to estimate the magnitude, location and amount of soft tissue strain induced by a combination of anatomic features of the brain and angular kinematics, and to provide an improved understanding of the mechanics of head injury.

3.6 COMPARISON OF OBLIQUE IMPACT KINEMATICS WITH FILM DATA

To compare simulated kinematics with actual test data, film data was reviewed and corresponding frames illustrating the impact sequence were assembled from film data and corresponding simulation results. Since sled-mounted door cameras provided the best film data for potential comparisons of impact kinematics, it was necessary to compensate for the sled velocity in order to generate and compare kinematic response data *relative* to the sled. Assuming the sled's side mounted cameras remained in a horizontal plane, this could be done by subtracting the sled velocity component from the headpart x and y velocity components. This was done only for the purpose of generating kinematic response which could be compared with sled side-mounted cameras *and does not replicate dynamic head loads* as was done above.

Figures 3-10a and 3-10b respectively illustrate the first half of a comparison of film data versus simulated kinematics as viewed from the camera mounted on the left or driver side of the occupant compartment. Figures 3-11a and 3-11b illustrate the second half of this sequence. An exact comparison of position and/or angular displacement versus time is very difficult to make because of uncertainties in reproducing identical viewing angles and effects of perspective. Large translational and angular displacements, complicated geometry and lack of clarity in the film data are also factors. However, despite these difficulties and uncertainties, it can be seen that the overall kinematics of the test are closely approximated by the simulation.

Similar results are seen from the kinematic comparison made using data taken from the passenger side sled-mounted camera shown in Figures 3-12a and 3-12b and Figures 3-13a and 3-13b. This comparison is sequenced in a similar manner with the first half of the test versus similar comparison shown in Figures 3-12a and 3-12b and the second in Figures 3-13a and 3-13b. Because the dummy is further away from the camera in this sequence, differences due to viewing perspective are probably smaller. Again, it appears that the overall kinematics are closely approximated by the simulation.

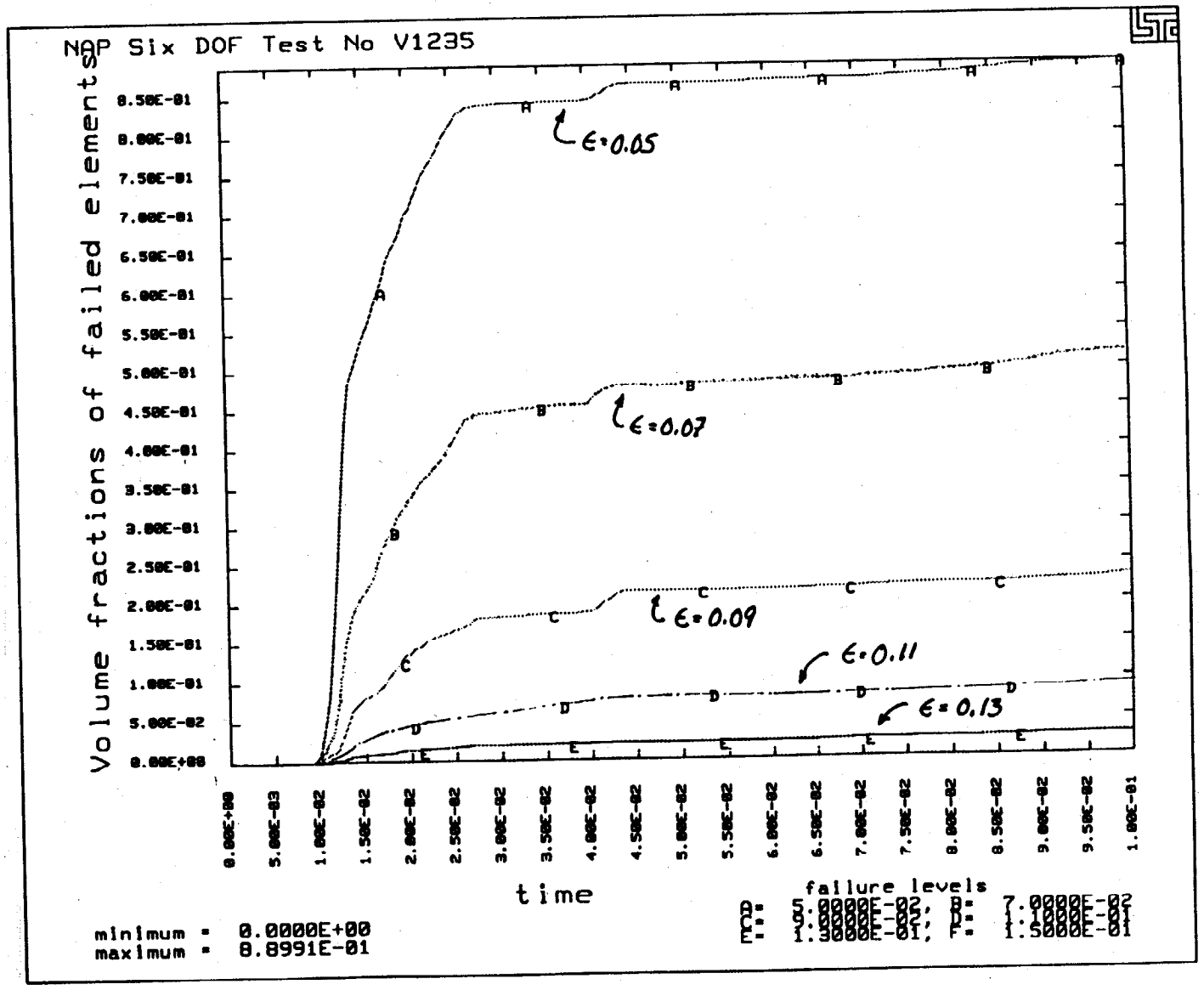


FIGURE 3-9. VOLUME FRACTION OF FAILED ELEMENTS VS STRAIN DEFORMABLE INTERIOR COMPONENT

3.7 RESULTANT ACCELERATIONS: OBLIQUE IMPACT SIMULATION VERSUS TEST

Head injury is currently computed according to the following Head Injury Criterion:

$$HIC = \left[1/(t_2 - t_1) \int_{t_1}^{t_2} a(t)dt \right]^{2.5} (t_2 - t_1) \qquad 3.1\text{-}1$$

where a(t) is the vector sum of acceleration components at the center of gravity of the dummy headpart (Bacc*i* *i*=1,3), and $(t_2 - t_1)$ is the time interval which maximizes HIC. The HIC value computed using measured accelerometer data in body coordinates was 854. Recomputing HIC using accelerations transformed to inertial coordinates provided a nearly identical HIC value of 852. Further, after using the velocity boundary conditions produced by the transformation process, applying them to the rigid headpart and extracting X, Y, and Z accelerations from the simulated response, a HIC value of 854 was obtained from the simulation based on the transformation process and resulting load curve data. Although the latter situation involves more computations and is potentially more prone to error, the results were essentially identical, as summarized below.

TABLE 3-1. SUMMARY OF HIC DATA

Data Source	HIC Value	t_1 (msec)	t_2 (msec)	t_2-t_1 (msec)
NAP (raw)	854	9.000	14.625	5.625
Transformed	852	8.875	14.750	5.875
Simulation	854	9.125	14.875	5.75

In both instances, HIC is virtually unchanged. These resultant acceleration time histories are shown in Figure 3-14, with HIC-related parameters contained in Table 3-1. From Figure 3-14, it can be seen that resultant acceleration time histories computed using measured NAP data versus accelerations, transformed to fixed coordinates, are virtually identical. Since the resultant acceleration should be invariant with the coordinate system in which it is computed, this provides an important check on the process.

In Figure 3-14, a comparison of resultant accelerations computed from (a) measurements made in the body coordinate system, and (b) accelerations transformed to inertial coordinates versus that produced from output accelerations computed after simulated headpart kinematics indicates only a minor change in amplitude and phase. These differences were not considered significant for the purpose of comparing headpart kinematics. If necessary, additional accuracy may be obtained in the simulation by increasing the number of points used to describe angular and translational velocity time

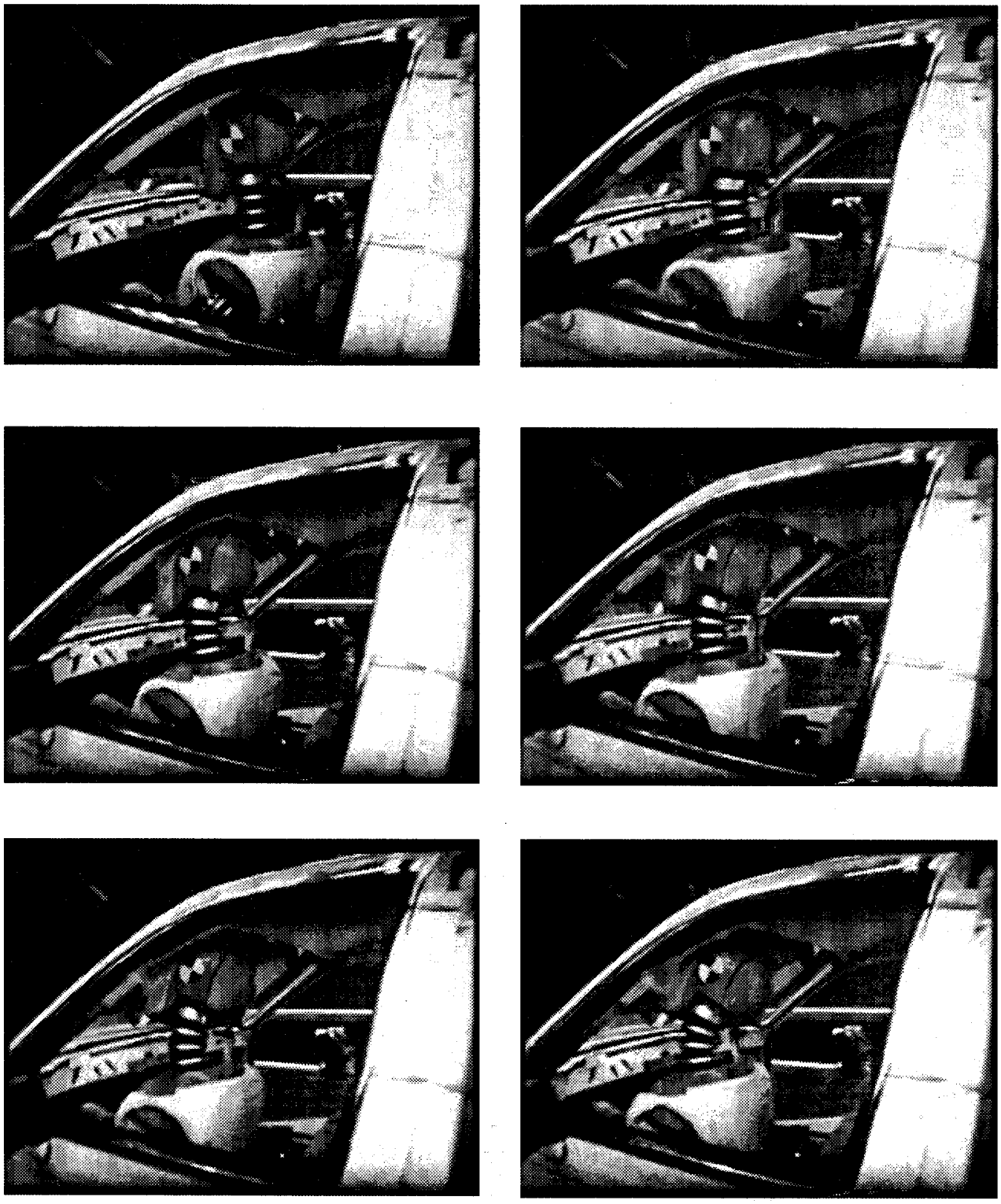


FIGURE 3-10A. FILM DATA FROM LEFT SIDE MOUNTED CAMERA (FIRST SEQUENCE)

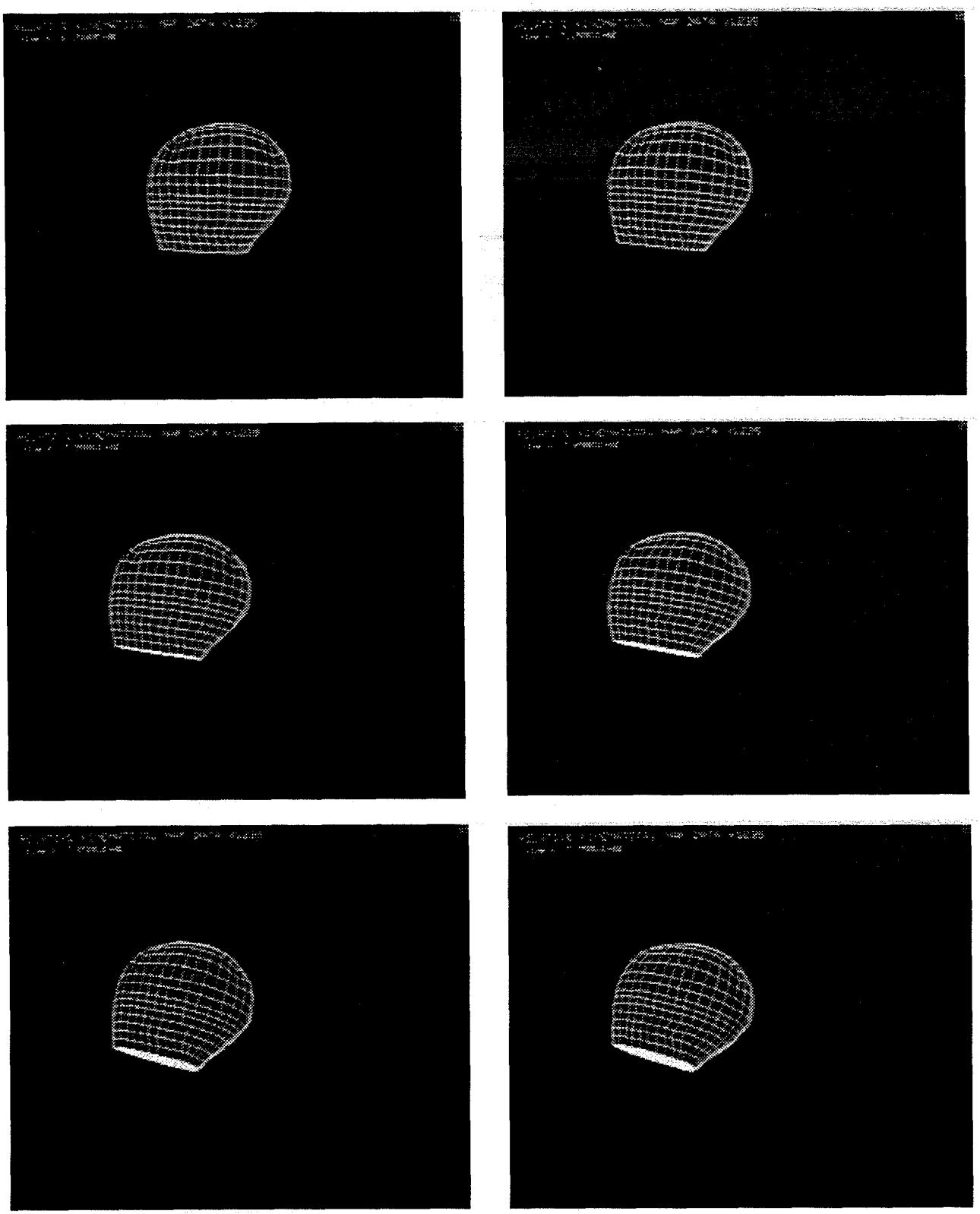


FIGURE 3-10B. SIMULATED KINEMATICS VIEWED FROM LEFT SIDE (FIRST SEQUENCE)

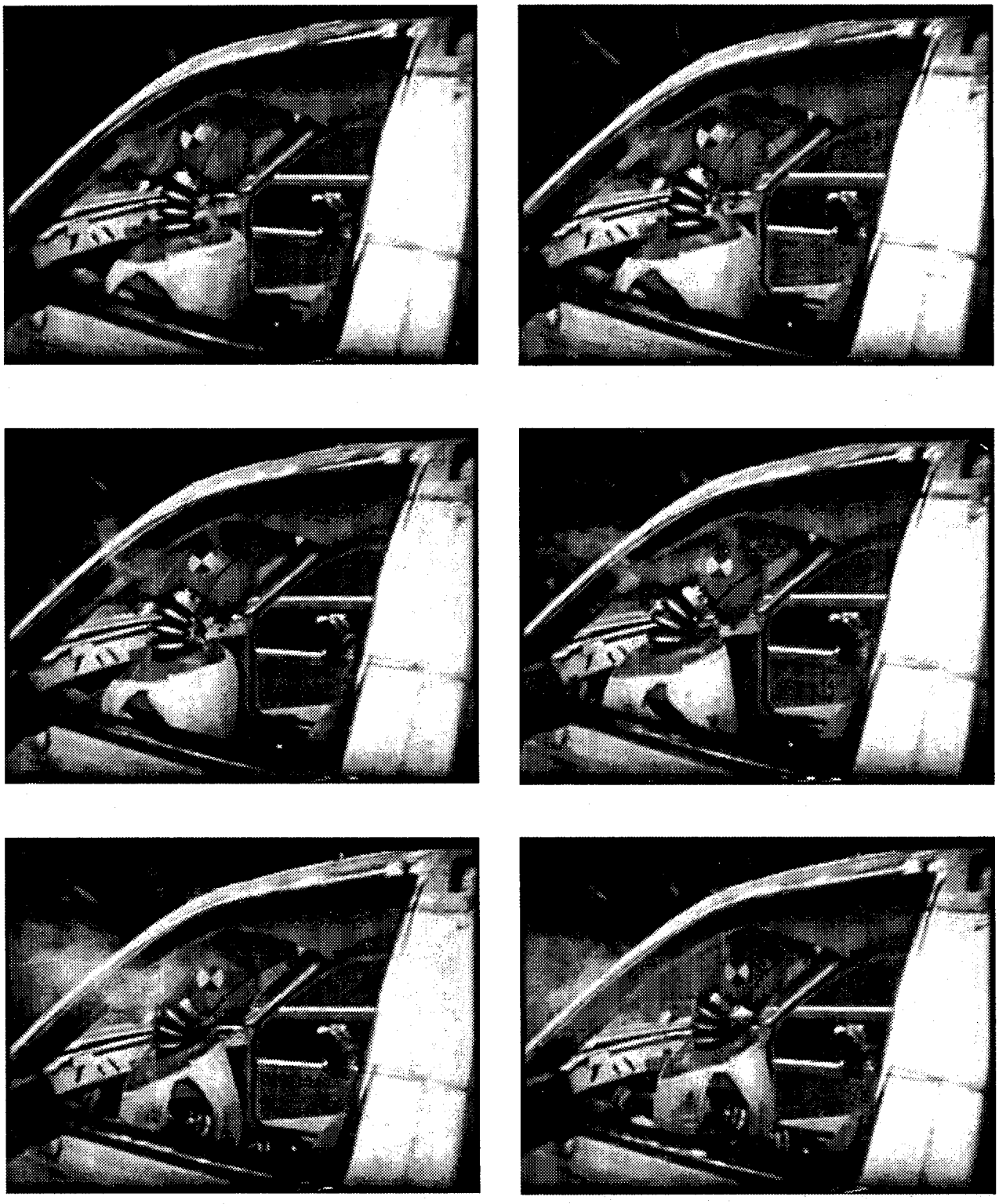


FIGURE 3-11A. FILM DATA FROM LEFT SIDE MOUNTED CAMERA (SECOND SEQUENCE)

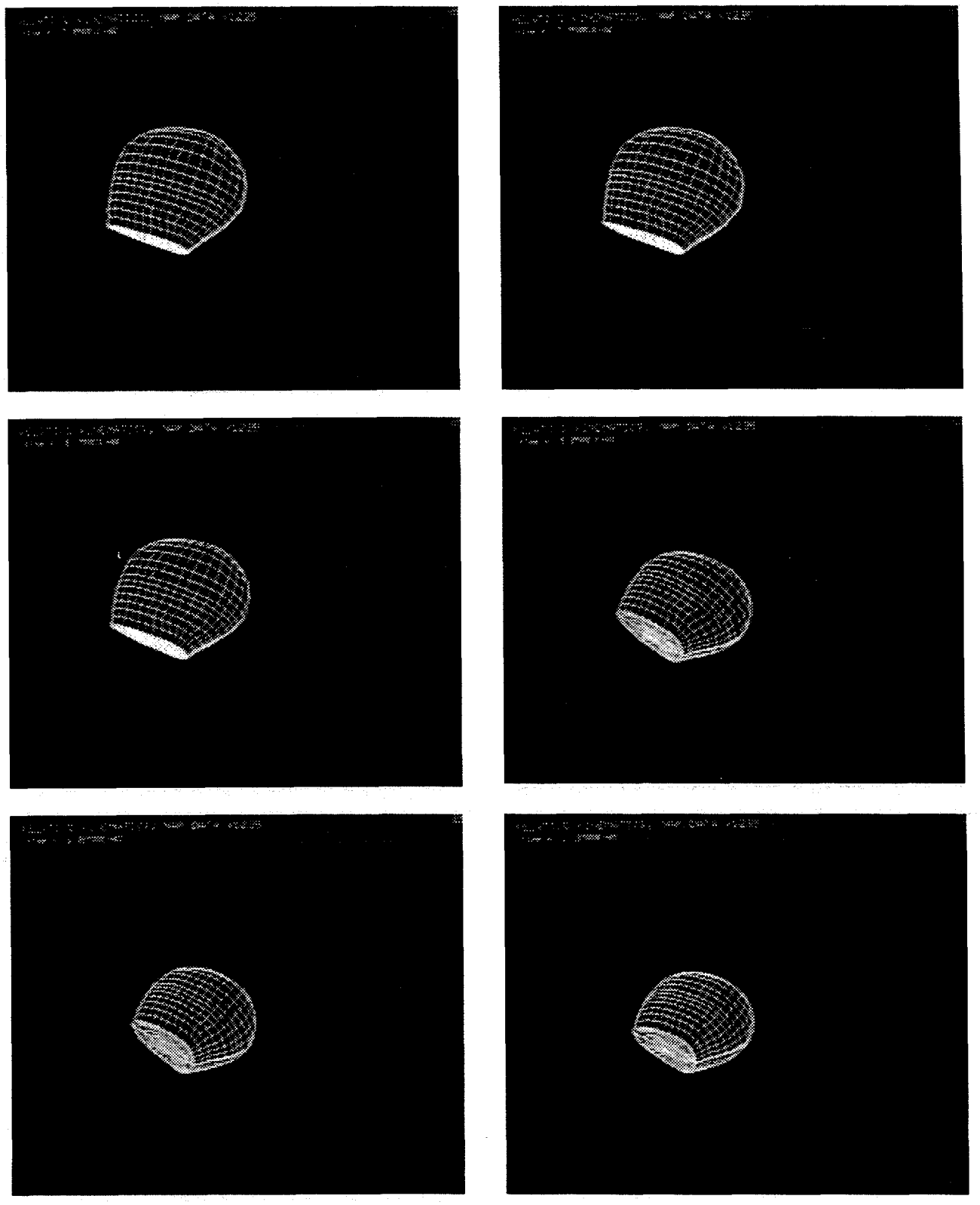


FIGURE 3-11B. SIMULATED KINEMATICS VIEWED FROM LEFT SIDE (SECOND SEQUENCE)

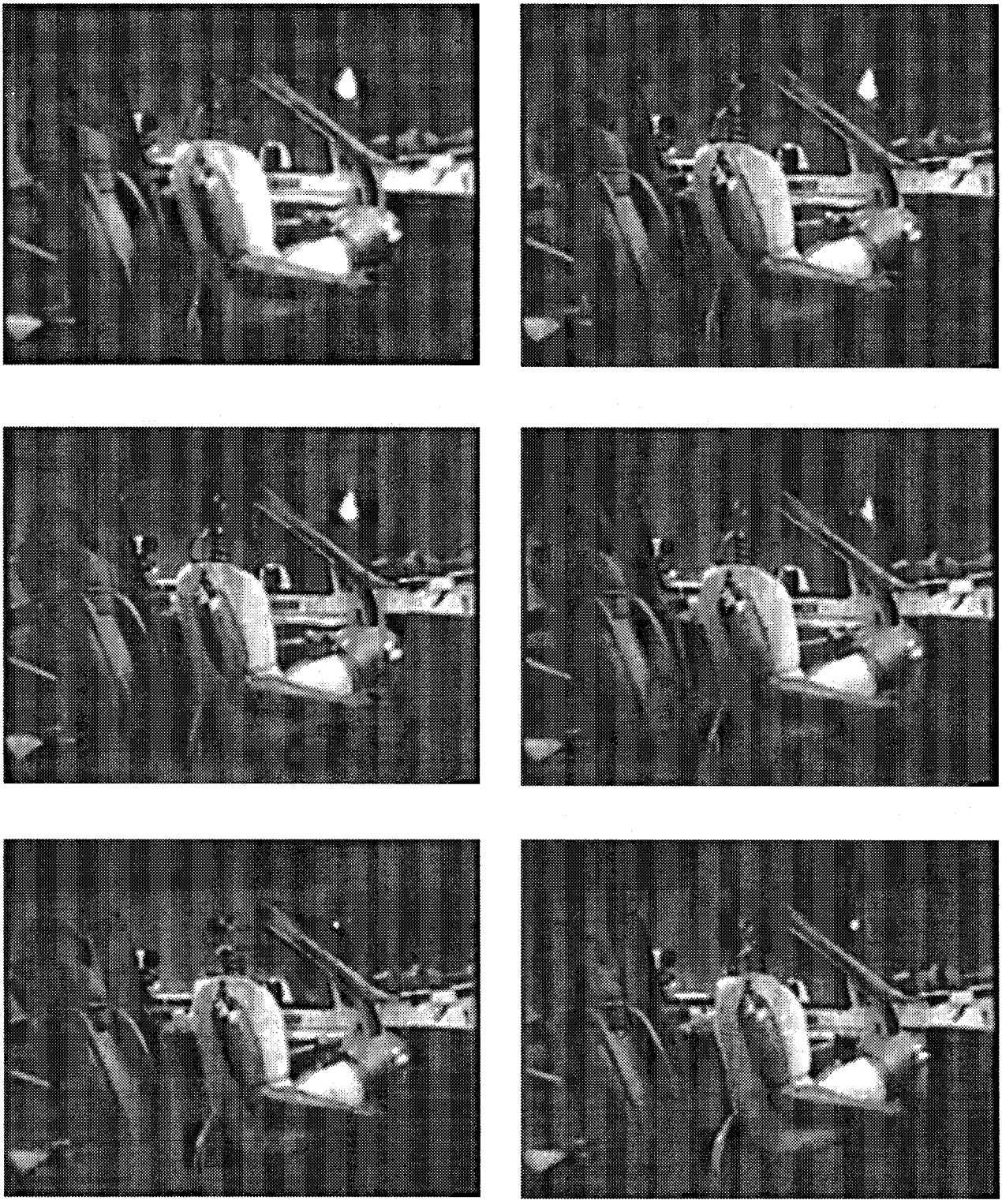


FIGURE 3-12A. FILM DATA FROM RIGHT SIDE MOUNTED CAMERA (FIRST SEQUENCE)

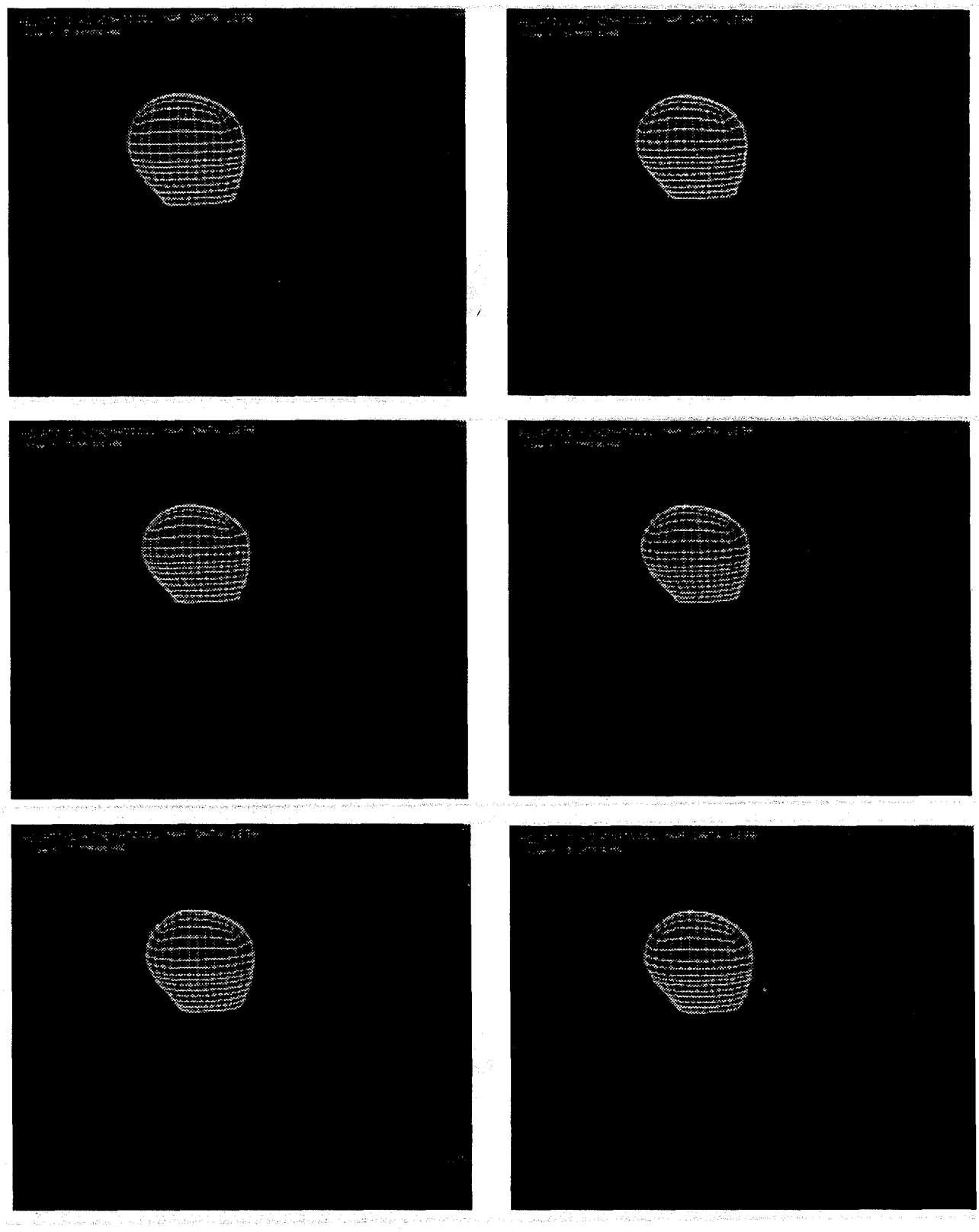


FIGURE 3-12B. SIMULATED KINEMATICS VIEWED FROM RIGHT SIDE (FIRST SEQUENCE)

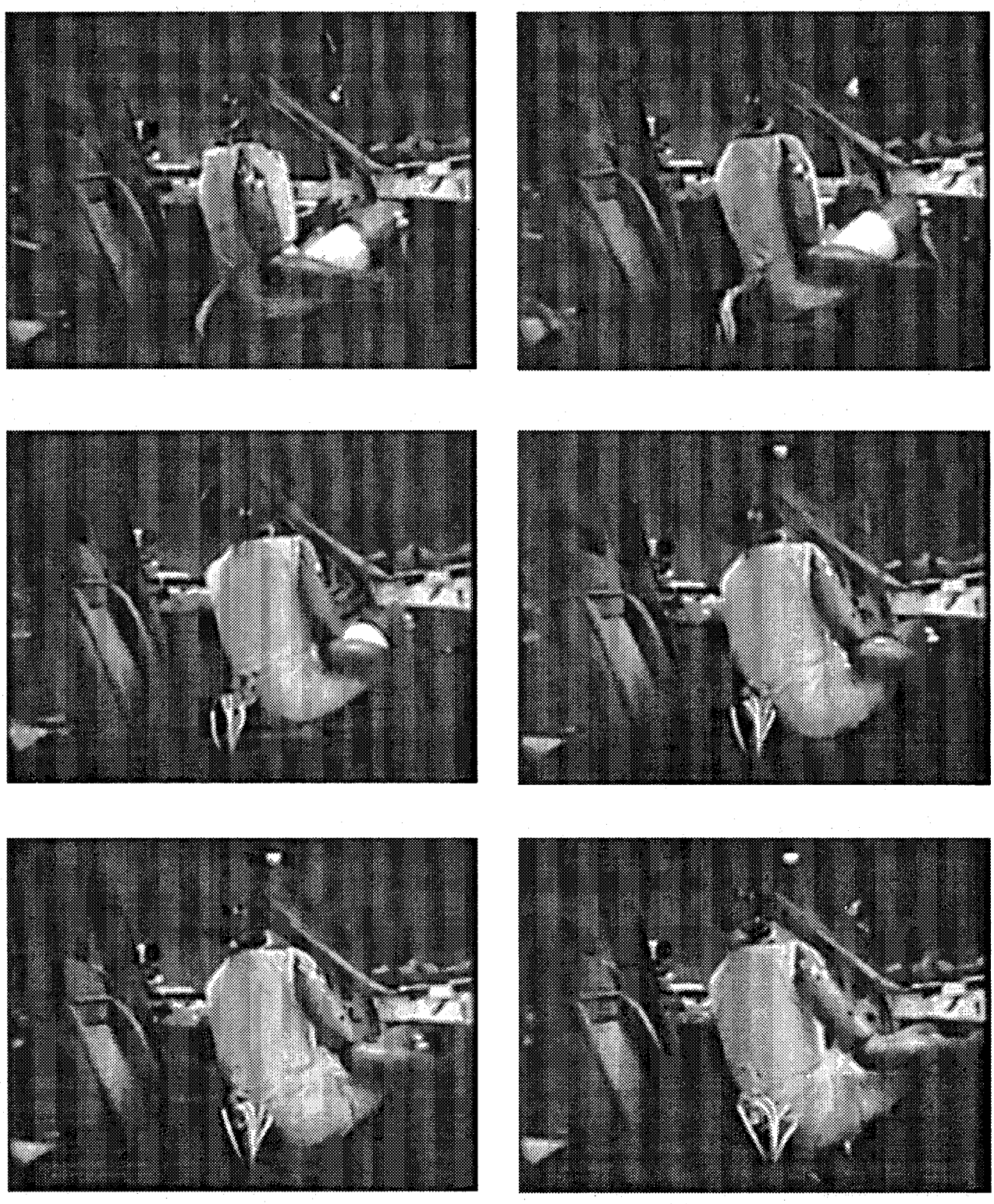


FIGURE 3-13A. FILM DATA FROM RIGHT SIDE MOUNTED CAMERA (SECOND SEQUENCE)

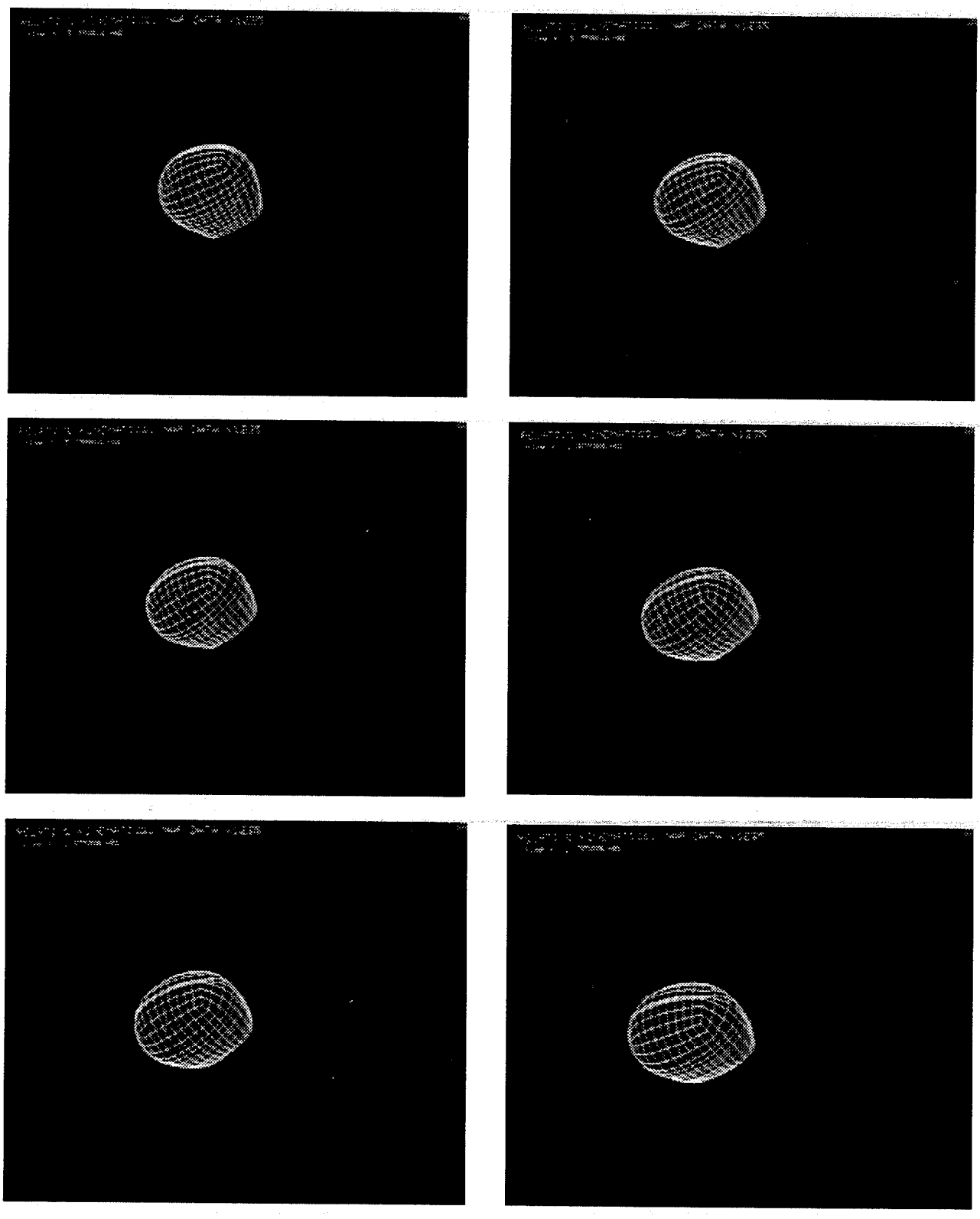


FIGURE 3-13B. SIMULATED KINEMATICS VIEWED FROM RIGHT SIDE (SECOND SEQUENCE)

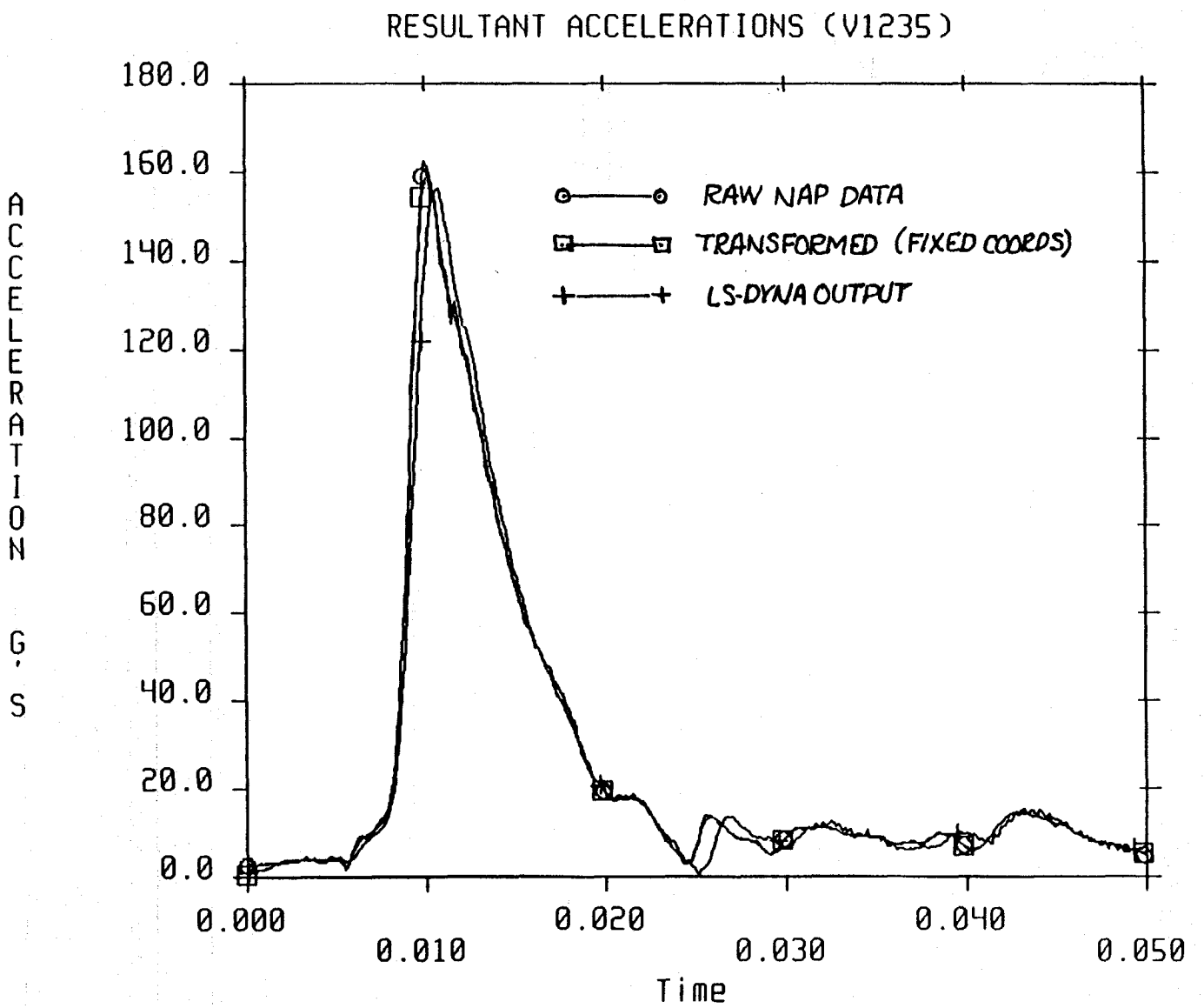


FIGURE 3-14. RESULTANT ACCELERATIONS BASED ON RAW, TRANSFORMED AND SIMULATION OUTPUT ACCELERATIONS

histories (i.e., the load curves), and/or increasing the number of significant decimal places used in generating velocity time histories.

3.8 TEMPORAL RELATIONSHIP BETWEEN STRAINS AND HIC

Referring to the failed volume and acceleration time histories of Figures 3-9 and 3-14, it can be seen that the percent volume of soft deformable interior material exceeding a specific strain level continues to rise, although at a reduced rate, well after the peak accelerations have occurred. The strain time-history data suggests additional deformation of the soft deformable component associated with post-impact angular kinematics. From Table 3-1, HIC is optimized between approximately 9 and 14.7 msec, with $(t_2 - t_1) = 5.7$ msec, while the accumulated volume of deformable material experiencing a particular level of strain has only reached approximately 67 percent of the final volume by that time. Depending on residual angular kinematics, the amount of total accumulated strain after the HIC clip time interval can be even larger.

Angular velocities about fixed X, Y, and Z axes are shown in Figure 3-15 for this data set and indicate that the dummy headpart continues to rotate well after accelerations at the CG have subsided as indicated by the resultant accelerations shown in Figure 3-14.

These data indicate that the temporal relationship between the accumulation of strain in soft interior components and the period in which HIC is optimized can be very different. HIC is computed over a relatively short period of time associated with large head accelerations at the CG, while the accumulation of strain in soft tissue occurs over a longer duration and is associated with angular kinematics which may continue after head impact. The implication of these results on assessing head injury is discussed in [2].

3.9 SUPPLEMENTAL KINEMATIC RESPONSE DATA - OBLIQUE IMPACT TEST

In addition to transformed angular velocity components shown in Figure 3-15, translational velocities at the CG, transformed to fixed coordinates, are also generated as shown in Figure 3-16, as are angular velocities about each axis in the body coordinate system as shown in Figure 3-17. Figures 3-15 and 3-16 define the generalized six-degree-of-freedom velocity profile of the headpart CG with respect to inertial coordinates. The X, Y, Z translational velocities time histories, followed by the X, Y, Z rotational velocity time histories, are sequentially written to the output file LC_FILE.DAT, and used to prescribe the kinematic motion of the headpart in the simulation process.

UDS FILES PRODUCED: ANGULAR VELOCITIES (FIXED COORDS.) - TEST 1235

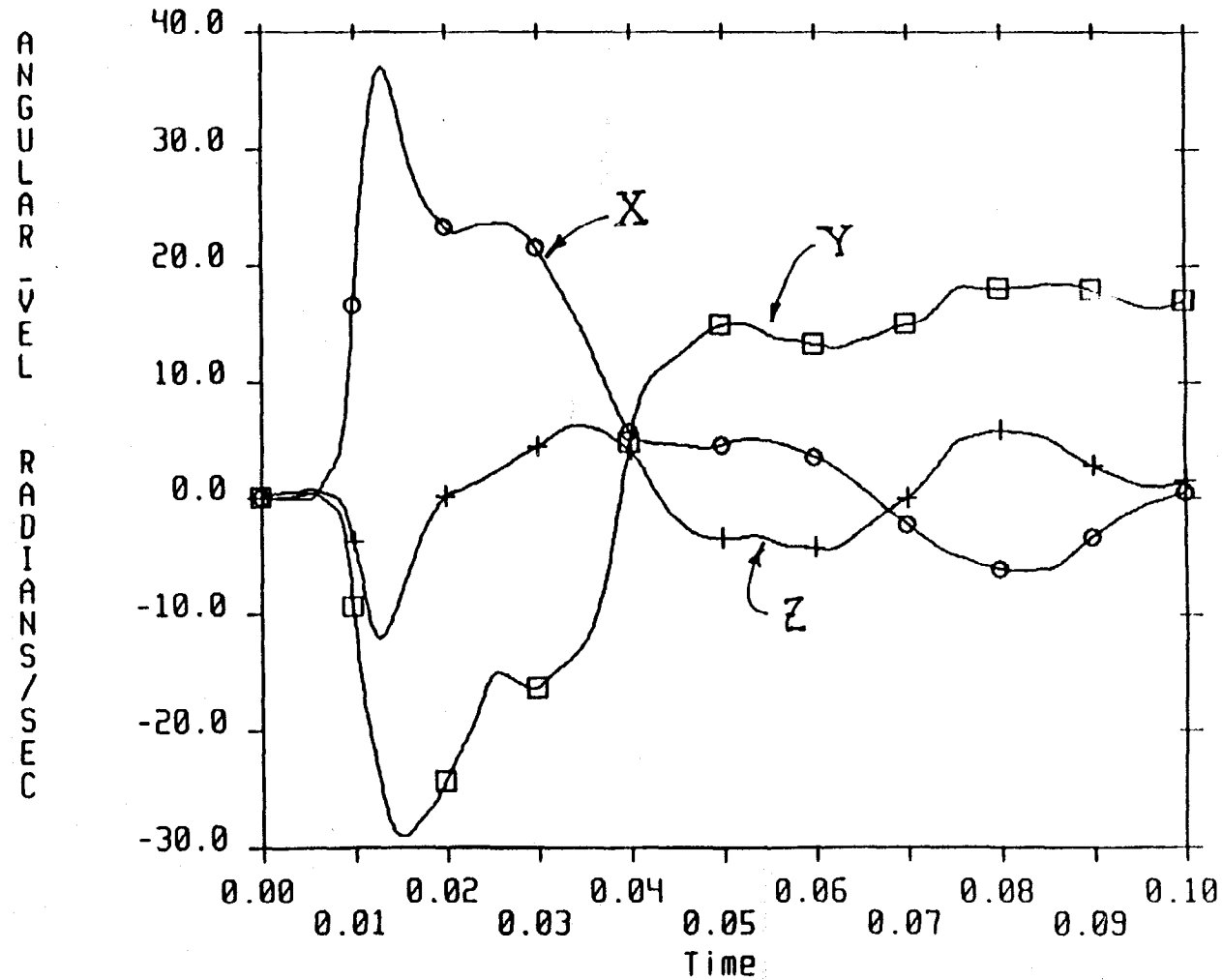


FIGURE 3-15. ANGULAR VELOCITIES TRANSFORMED TO FIXED COORDINATES OBLIQUE IMPACT TEST

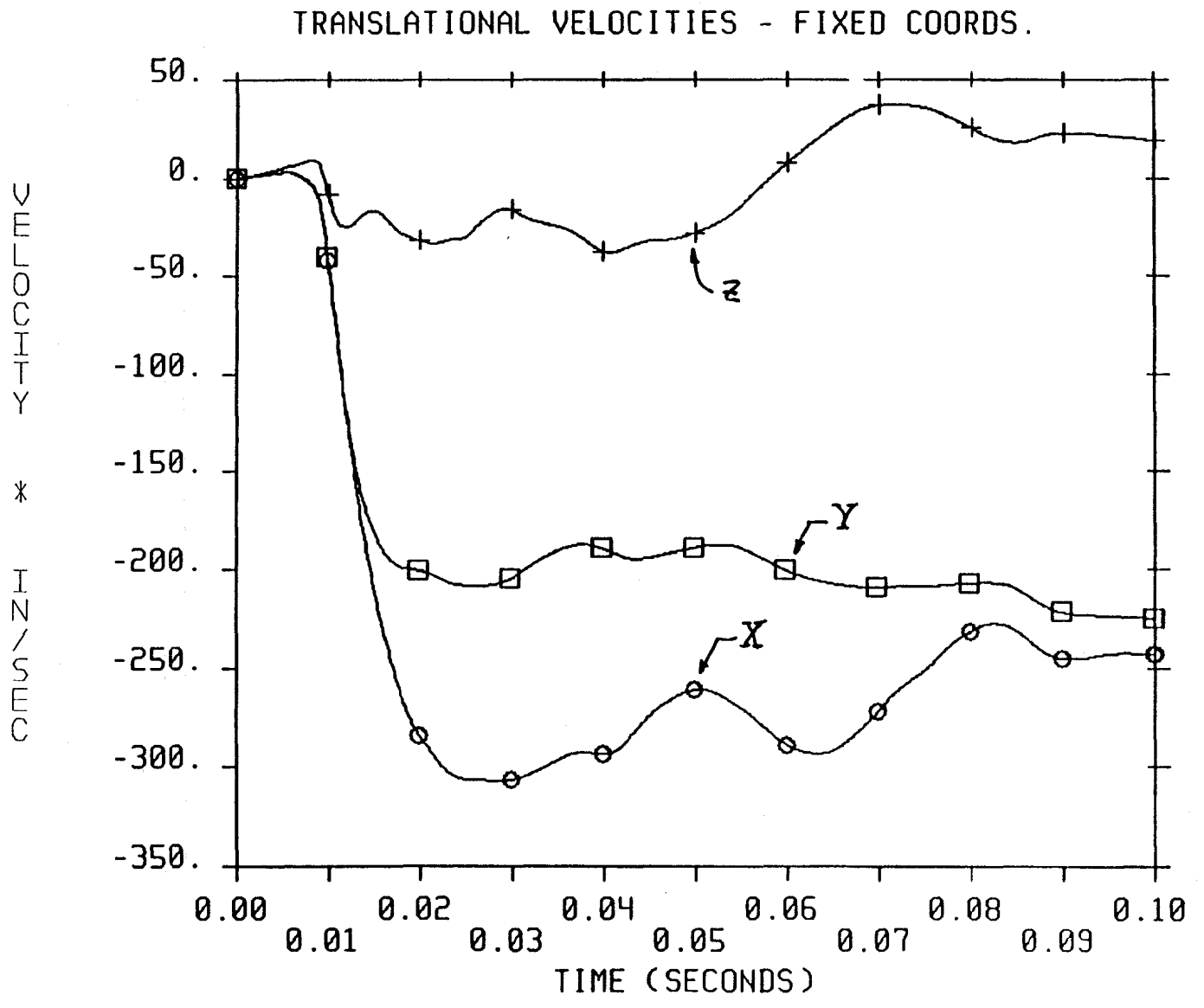


FIGURE 3-16. TRANSLATIONAL VELOCITIES TRANSFORMED TO FIXED COORDINATES OBLIQUE IMPACT TEST

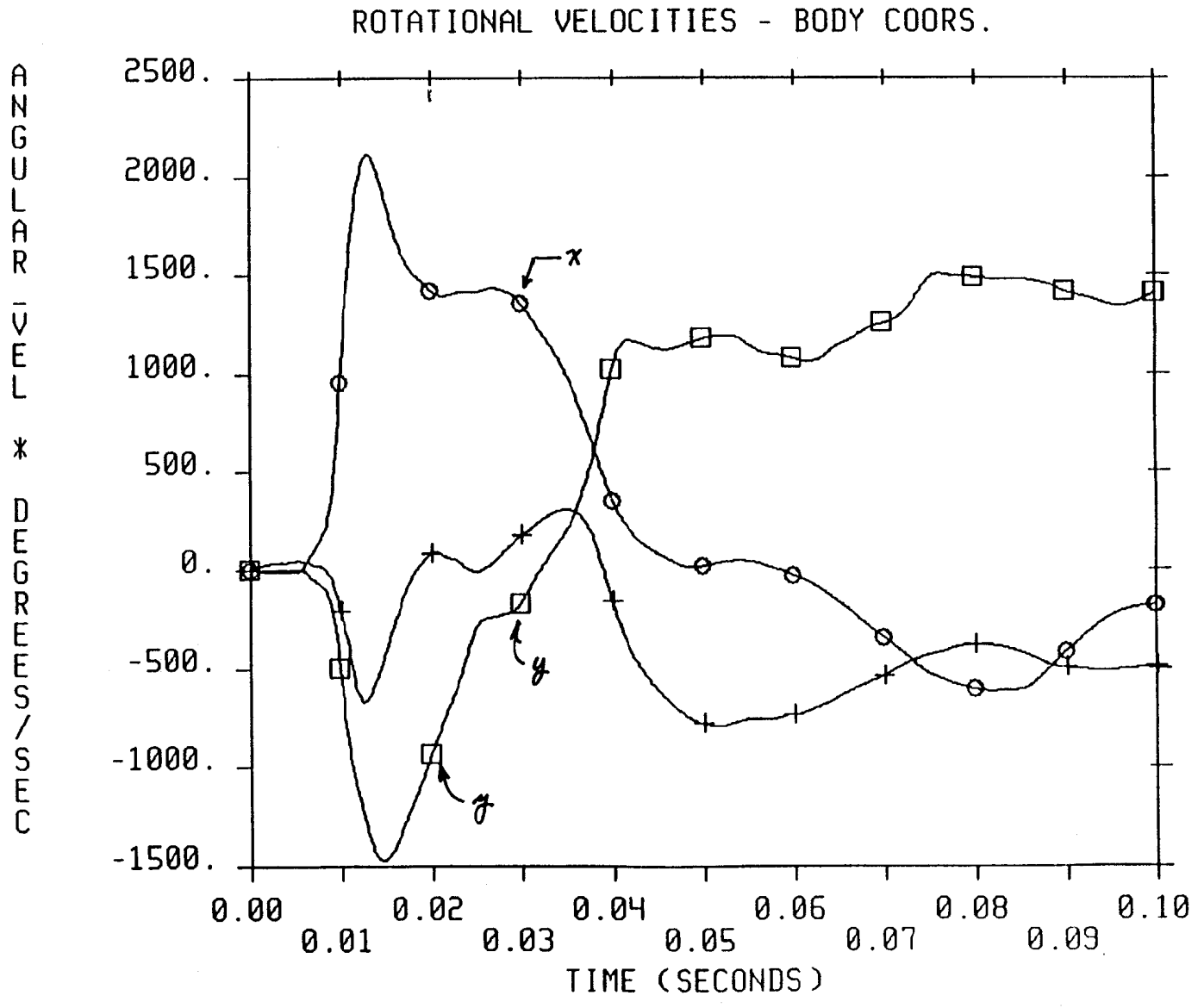


FIGURE 3-17. ANGULAR VELOCITIES IN BODY COORDINATE SYSTEM OBLIQUE IMPACT TEST

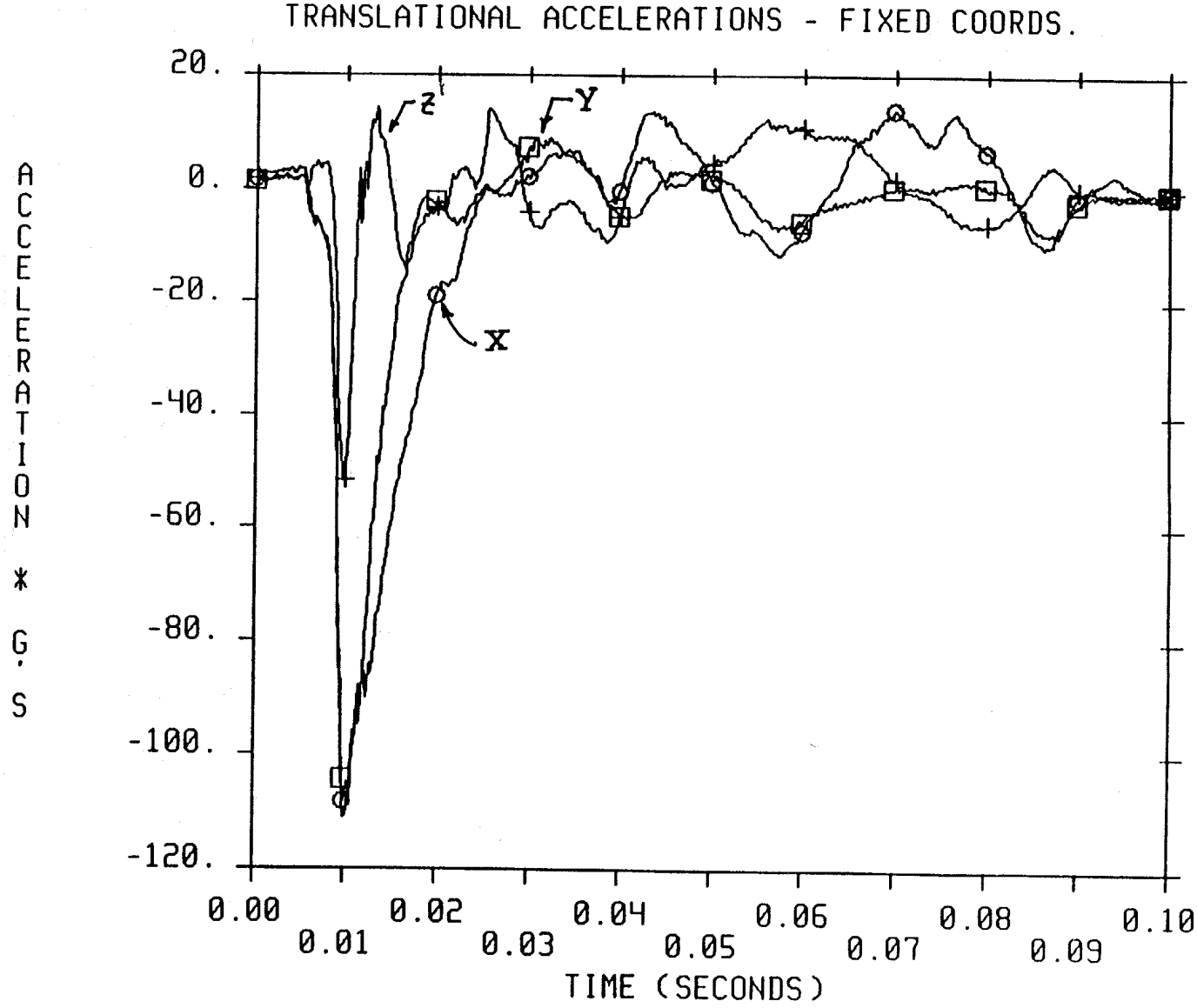


FIGURE 3-18. TRANSLATIONAL ACCELERATIONS IN FIXED COORDINATES

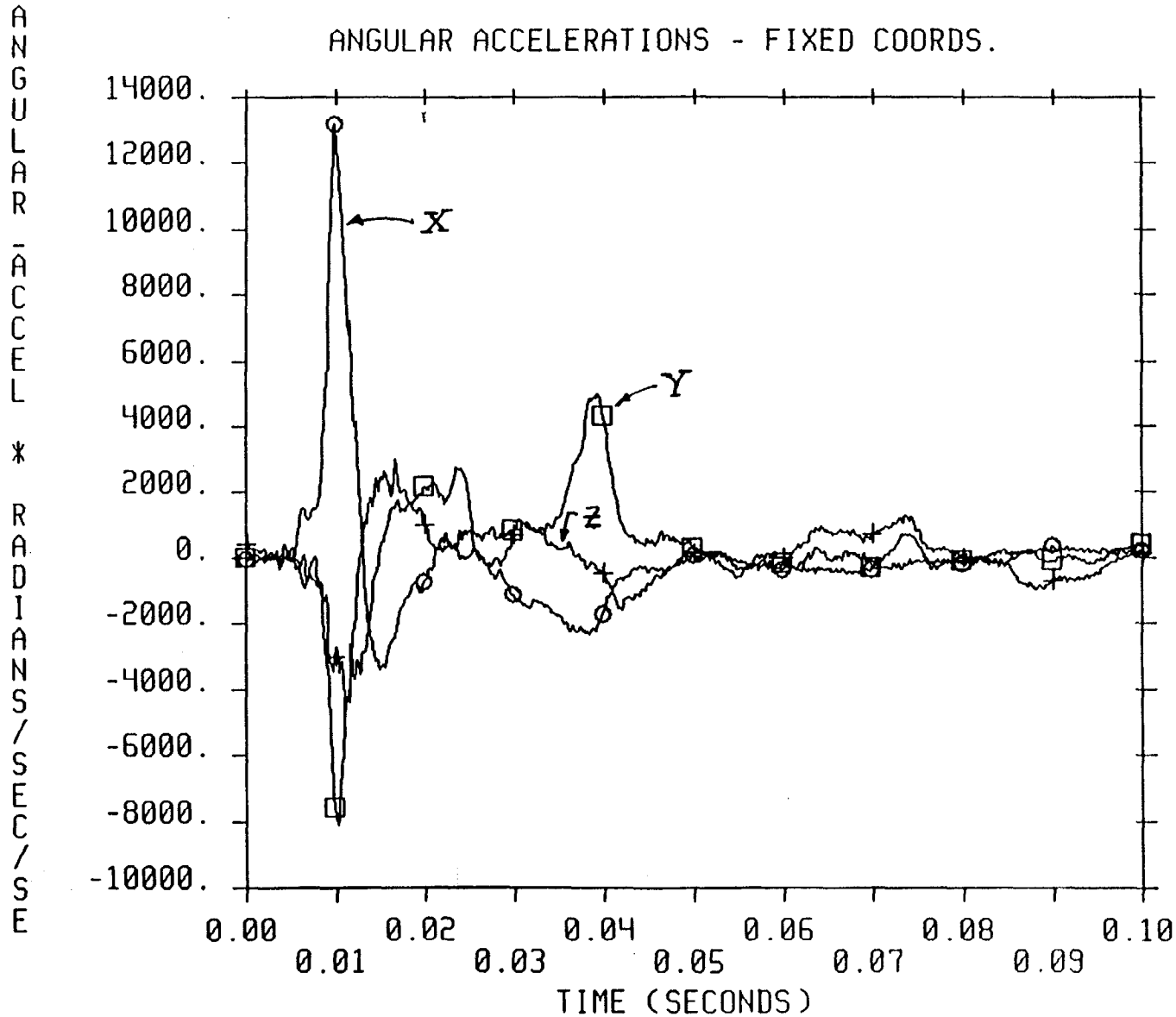


FIGURE 3-19. ROTATIONAL ACCELERATIONS IN FIXED COORDINATES

Translational and rotational accelerations at the headpart CG, transformed to fixed coordinates, are shown in Figures 3-18 and 3-19, respectively. Acceleration data from Figure 3-18 was used to compute the resultant acceleration relative to fixed coordinates as shown in Figure 3-14, while accelerometer data at the CG of the NAP array was used to compute the corresponding resultant acceleration in body coordinates.

Elements of the updated direction cosine matrix relating angular orientation between principal body and fixed coordinates are illustrated in Figure 3-20, and angular displacements with respect to the fixed coordinate system are illustrated in Figure 3-21. The latter data may be used to define the angular position of the headpart at any point in time. For example, from Figure 3-21 it can be seen that pitch angular displacement peaks at about 40 msec and that the angular displacements of the headpart with respect to the global X, Y, and Z axes are about 43.5, -26, and -10 degrees, respectively, at this time.

Using post-processing software, the headpart can be rotated into its angular position at 40 msec by using the angular displacement data from Figure 3-21, and for comparison, overlaid with the headpart angular position taken from simulated kinematic response data at the corresponding instant. This is illustrated in Figure 3-22. It can be seen that the angular orientation is virtually identical. This view is a left elevation view and is comparable to latter stages of the film data sequence shown in Figures 3-10a and 3-11a. (Again, differences in viewing angle and perspective exist between test and simulated data). An overlay of angular orientation as viewed from a frontal or anterior vantage point, is shown in Figures 3-23 at the 40 msec state. Similar overlays at 100 msec are shown in Figures 3-24 and 3-25 for lateral elevation and frontal views, respectively, using headpart angular displacements about X, Y, and Z axes of 43.5, -26 and -10 degrees, respectively, taken from Figure 3-21 at the end of the simulation.

The above examples illustrate how supplemental kinematic response data produced during the transformation process may be used to illustrate headpart orientation and angular kinematics which are not always easily determined or discernible from film data. These data could be very useful in studying headpart kinematics from crash tests where airbag deployment obscures and/or prevents the observance of headpart kinematics from either vehicle-mounted or ground-based cameras or in cases where film data is not sufficient for assessing headpart kinematics.

3.10 SIDE IMPACT SIMULATION OF HEAD TO B-PILLAR CONTACT

To provide a good kinematic comparison with film data having reduced distortion due to perspective, and to simulate a generically different impact scenario for demonstration, a side impact test was selected where the head struck the vehicle B-pillar structure. The B-pillar is located rearward of the A-pillar, supporting the vehicle roof and door assembly structures and is typically located just to the rear of a seated driver in a four-door vehicle.

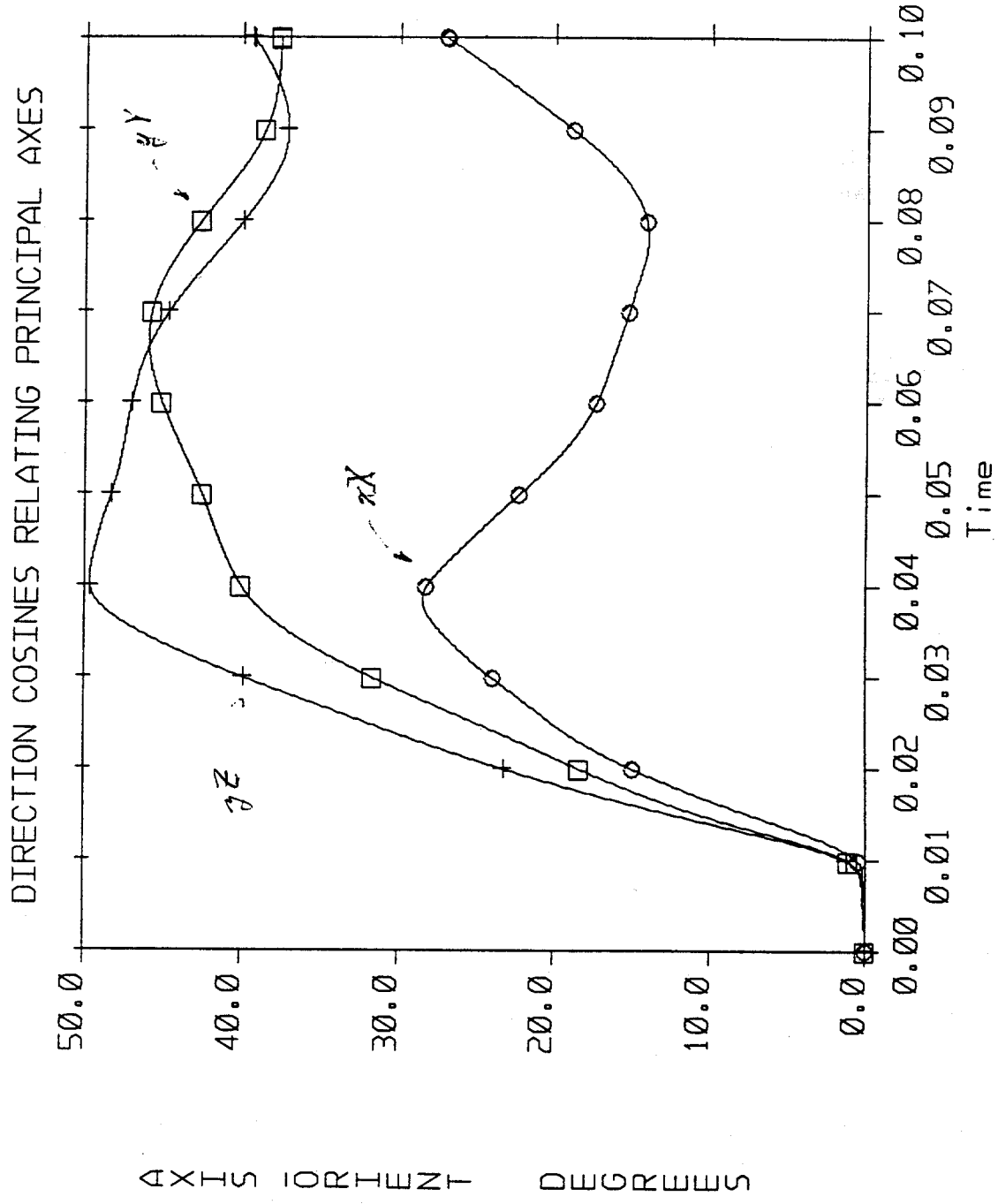


FIGURE 3-20. DIRECTION COSINES RELATING PRINCIPAL AXES

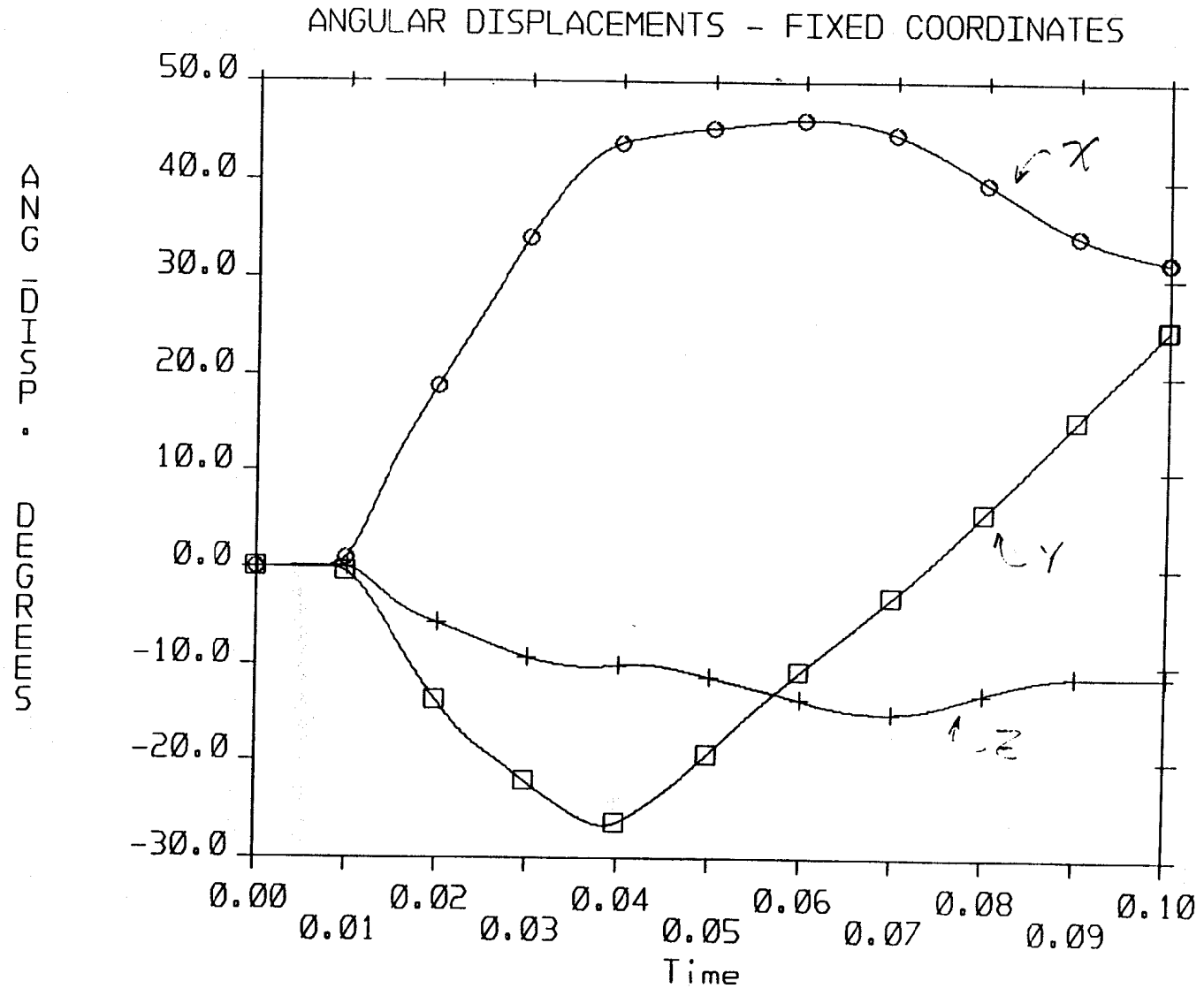


FIGURE 3-21. ANGULAR DISPLACEMENTS RELATIVE TO FIXED COORDINATES

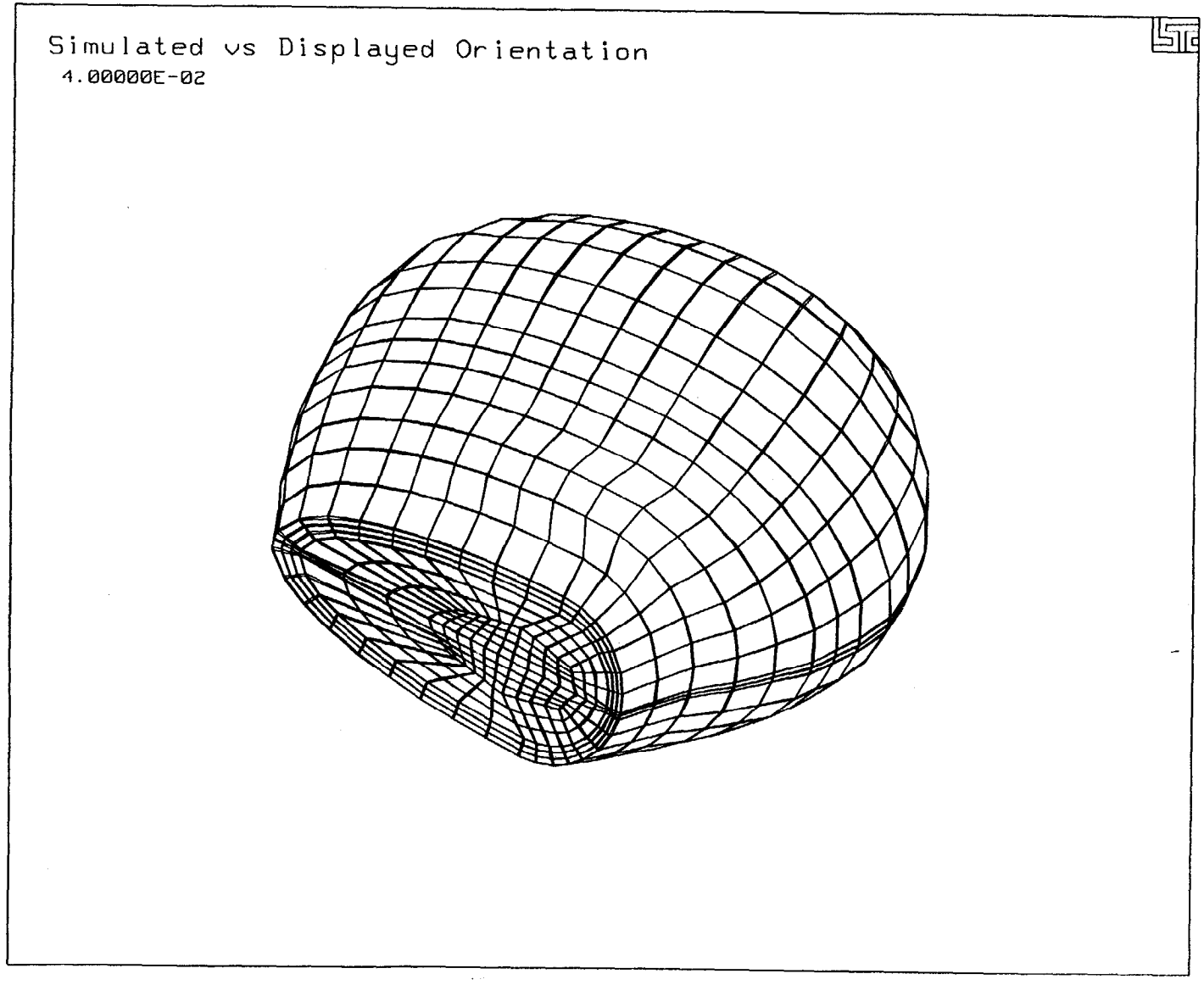


FIGURE 3-22. OVERLAY OF HEADPART POSITION AT 40 MSEC - LEFT ELEVATION VIEW

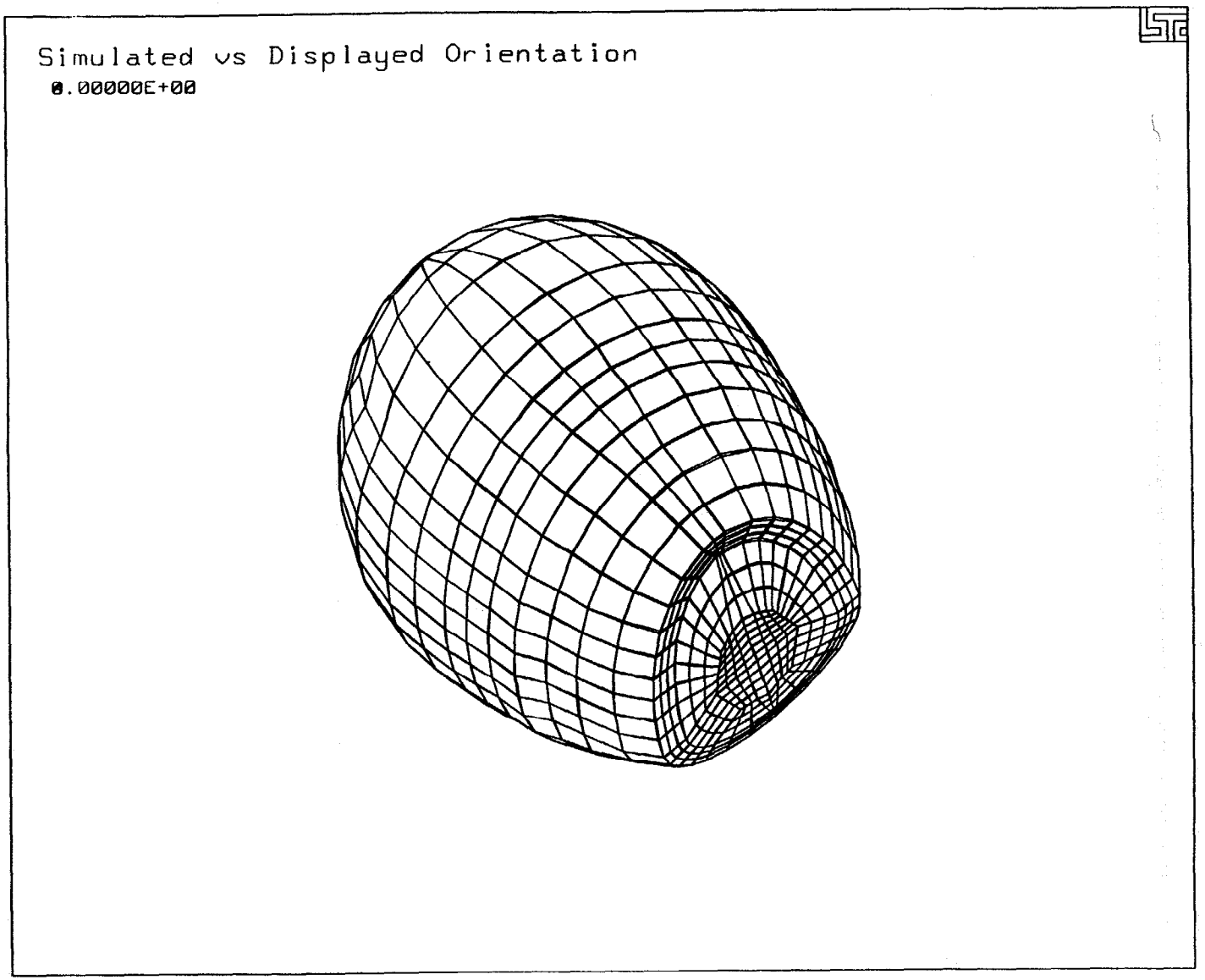


FIGURE 3-23. OVERLAY OF HEADPART POSITION AT 40 MSEC - FRONT VIEW

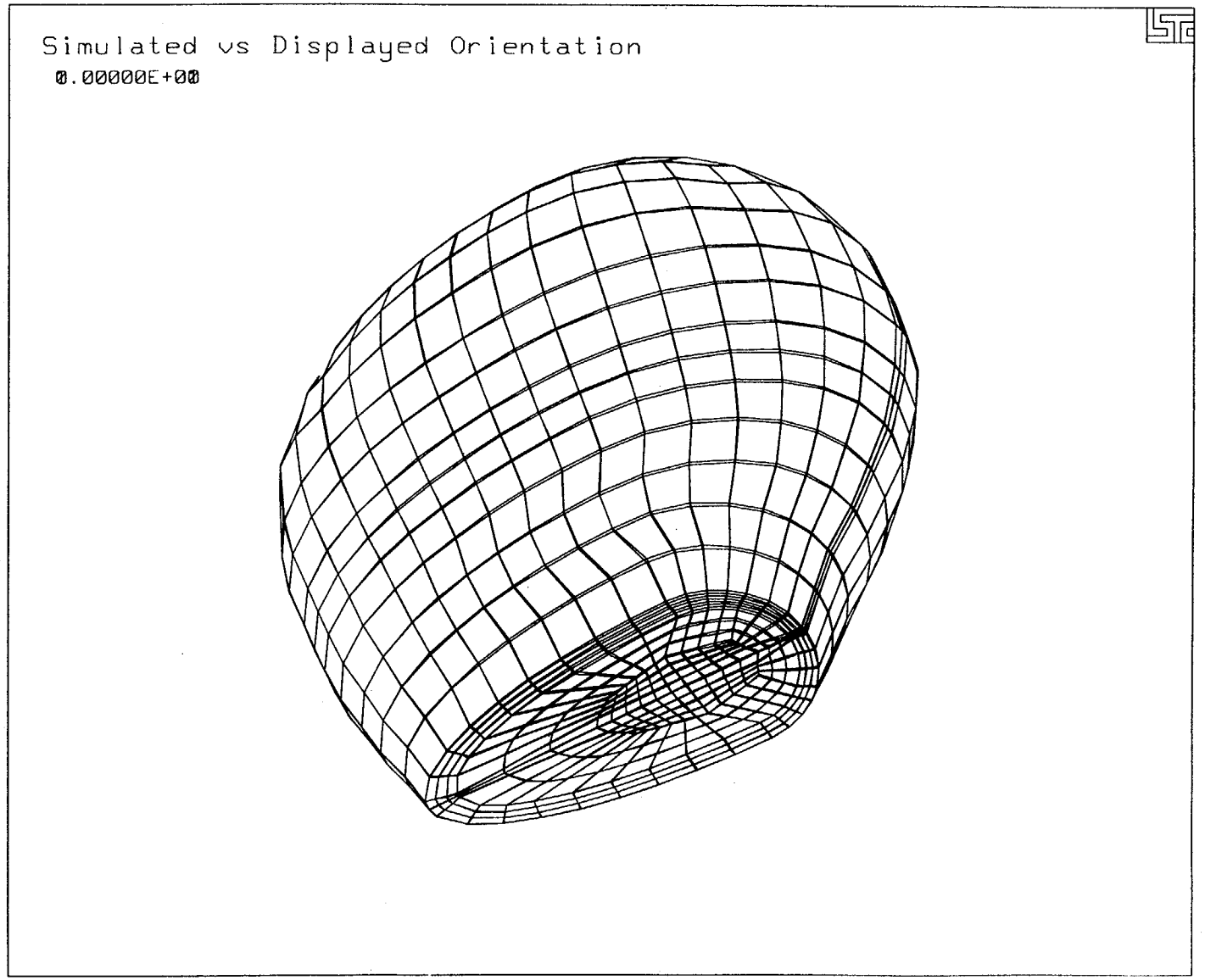


FIGURE 3-24. OVERLAY OF HEADPART POSITION AT 100 MSEC - LEFT ELEVATION VIEW

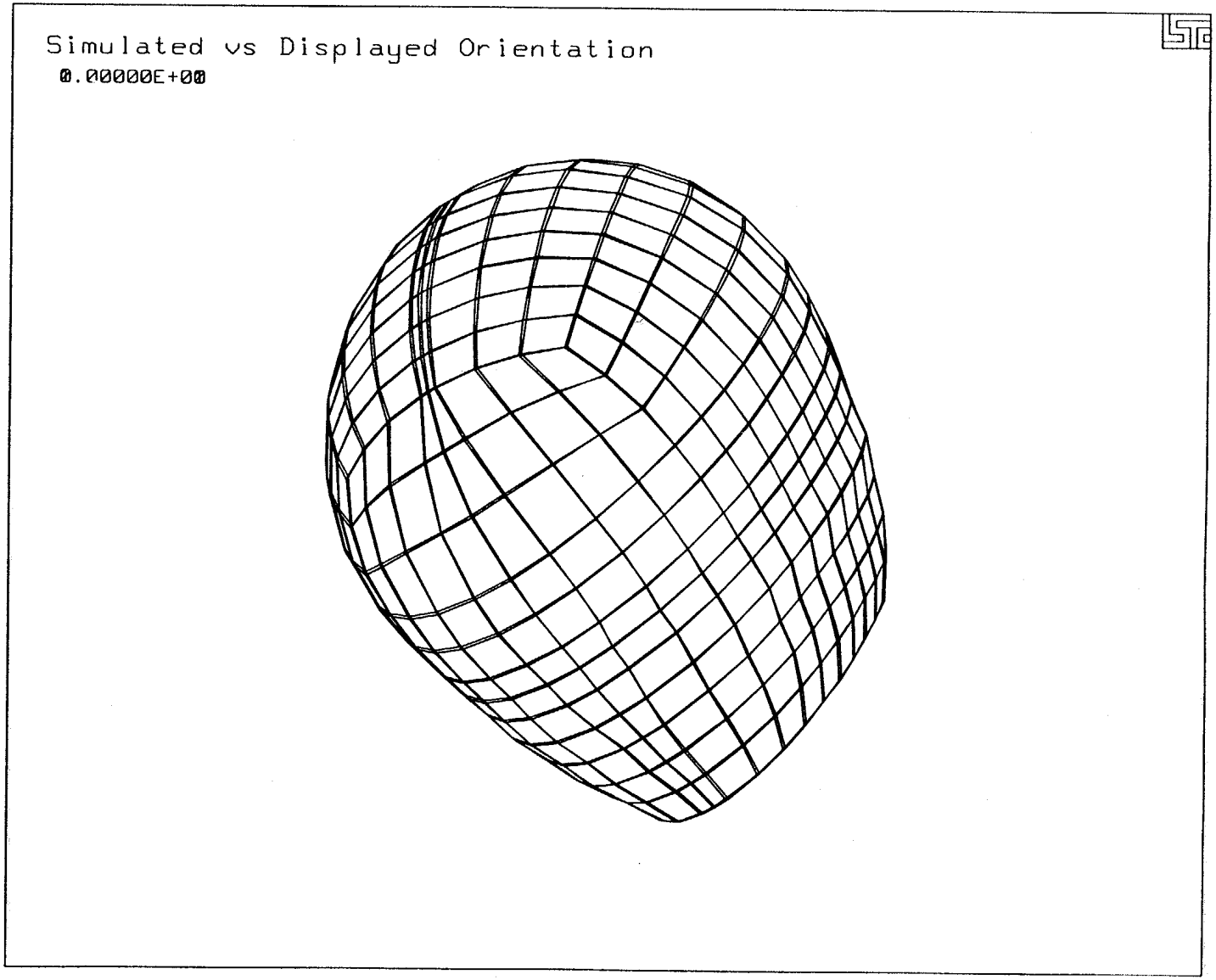


FIGURE 3-25. OVERLAY OF HEADPART POSITION AT 100 MSEC - FRONT VIEW

In this impact scenario, the dummy's shoulder region is closest to the vehicle side and is likely to make first contact with the side structure, causing the head to lean or roll accordingly. At a later time, the posterior region of the head is likely to contact the B-pillar, inducing a yaw motion of the headpart. Again, kinematic response should remain primarily in the lateral plane due to head-neck interaction with the lower dummy structure.

After reviewing available test data and screening accelerometer data as described for the previous simulation, another 1984 Ford Tempo was selected (NHTSA test number v1236). From the test report documentation, this vehicle was a four door sedan, placed on the sled at an angle of 90 degrees clockwise, with respect to the sled axis (similar to Figure 3-2) to produce a 15 mph head impact with the driver side B-pillar, which was again lined with production trim. This test was another in a series of tests to compare head impact response using production trim versus special padding.

An unrestrained Alderson 50th percentile HYBRID III dummy was located at the left front driver position, and head contact with the B-pillar was noted in the test report, to produce a HIC value of 791. Data channels corresponding to NAP data were labeled channels 4 through 12 and these were reviewed for quality and prepared in the same manner as discussed for the previous test, with a time shift of 50 msec. Once again, when comparing simulated kinematics with film data from sled mounted door cameras, headpart kinematics were adjusted to compensate for sled kinematics. This is done only for the purpose of comparing relative kinematics, and *does not* replicate dynamic head loading.

Figure 3-26 is an overlay of a rear elevation view of simulated headpart kinematics for this impact event in 20 msec intervals, where the vehicle is being accelerated into the dummy from left to right (i.e., in the negative Y direction). An overlay of the corresponding frontal view is shown in Figure 3-27, also in 20 msec intervals. As appropriate for this test, the headpart moves laterally across the vehicle in the direction of the sled velocity vector, moving in a near horizontal plane. The more subtle headpart motions which occur between 20 and 30 msec are difficult to capture with figures, but are clearly discernible when viewed or played back as a "movie" with all 200 states displayed on a workstation.

The tiles of Figures 3-28 and 3-29 have been assembled to describe headpart motion, relative to headpart initial position, and to illustrate the more subtle rolling and yawing motions of the headpart caused by shoulder-side structure contact and subsequent head B-pillar contact, respectively. In each figure, the solid lines illustrates motion of the headpart and the dashed lines indicate headpart initial position (i.e., stationary with respect to the vehicle). Headpart kinematics between 17 and 25 msec are illustrated from a posterior view in Figure 3-28, with a scale factor of two applied to the displacement field, in order to illustrate the small motions involved. In the second and third tiles of

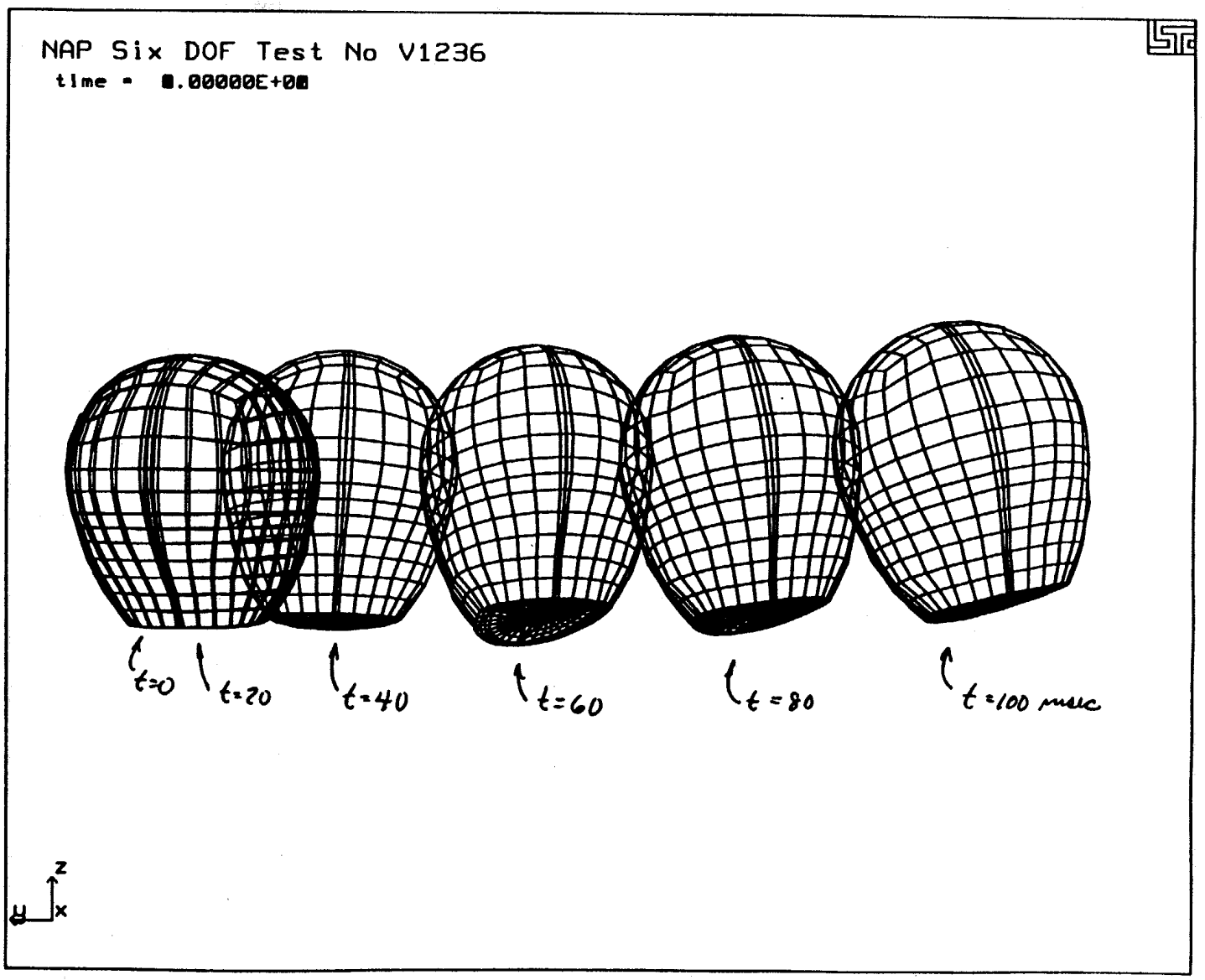


FIGURE 3-26. REAR ELEVATION VIEW OF KINEMATIC RESPONSE - SIDE IMPACT TEST

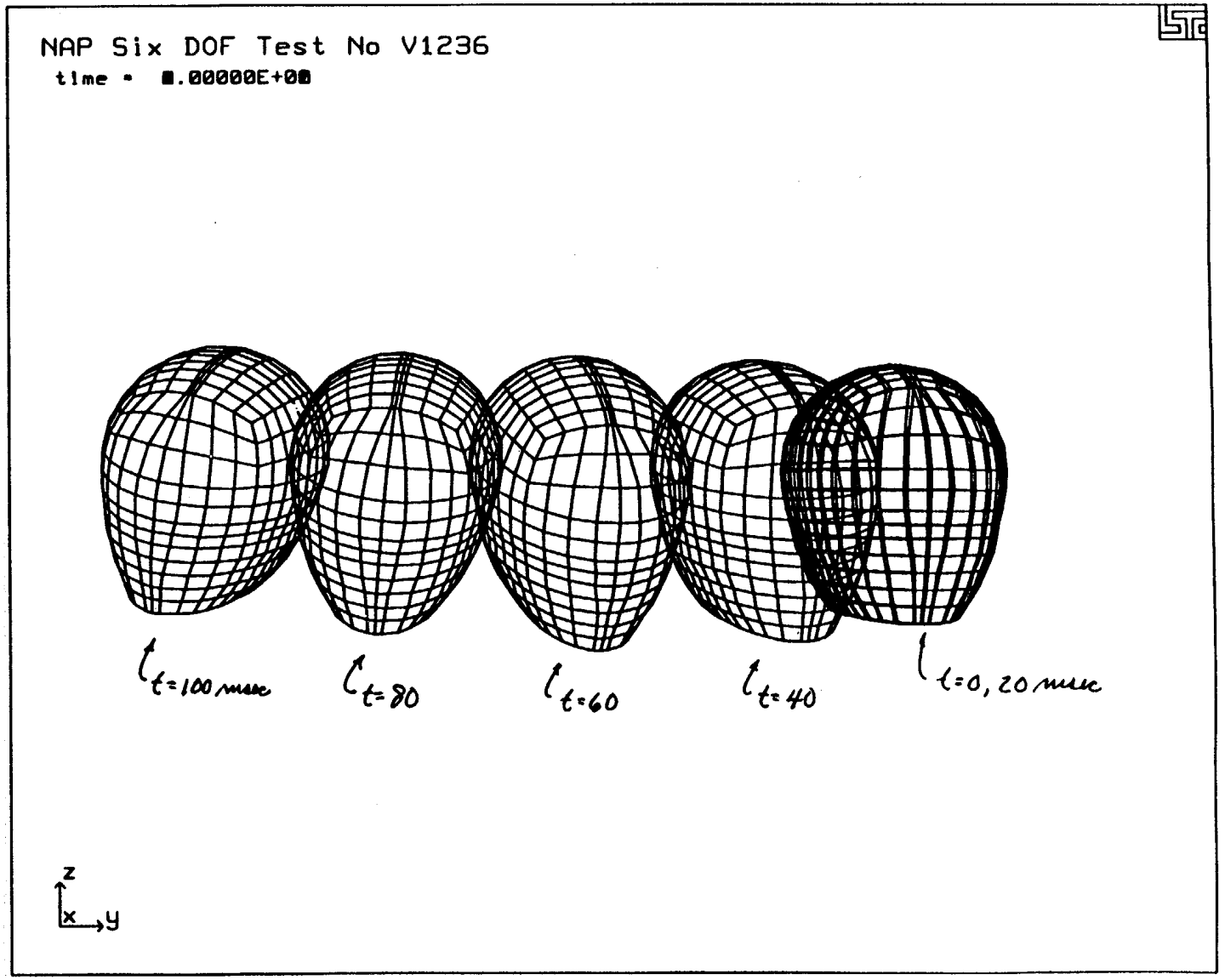


FIGURE 3-27. FRONT ELEVATION VIEW OF KINEMATIC RESPONSE - SIDE IMPACT TEST

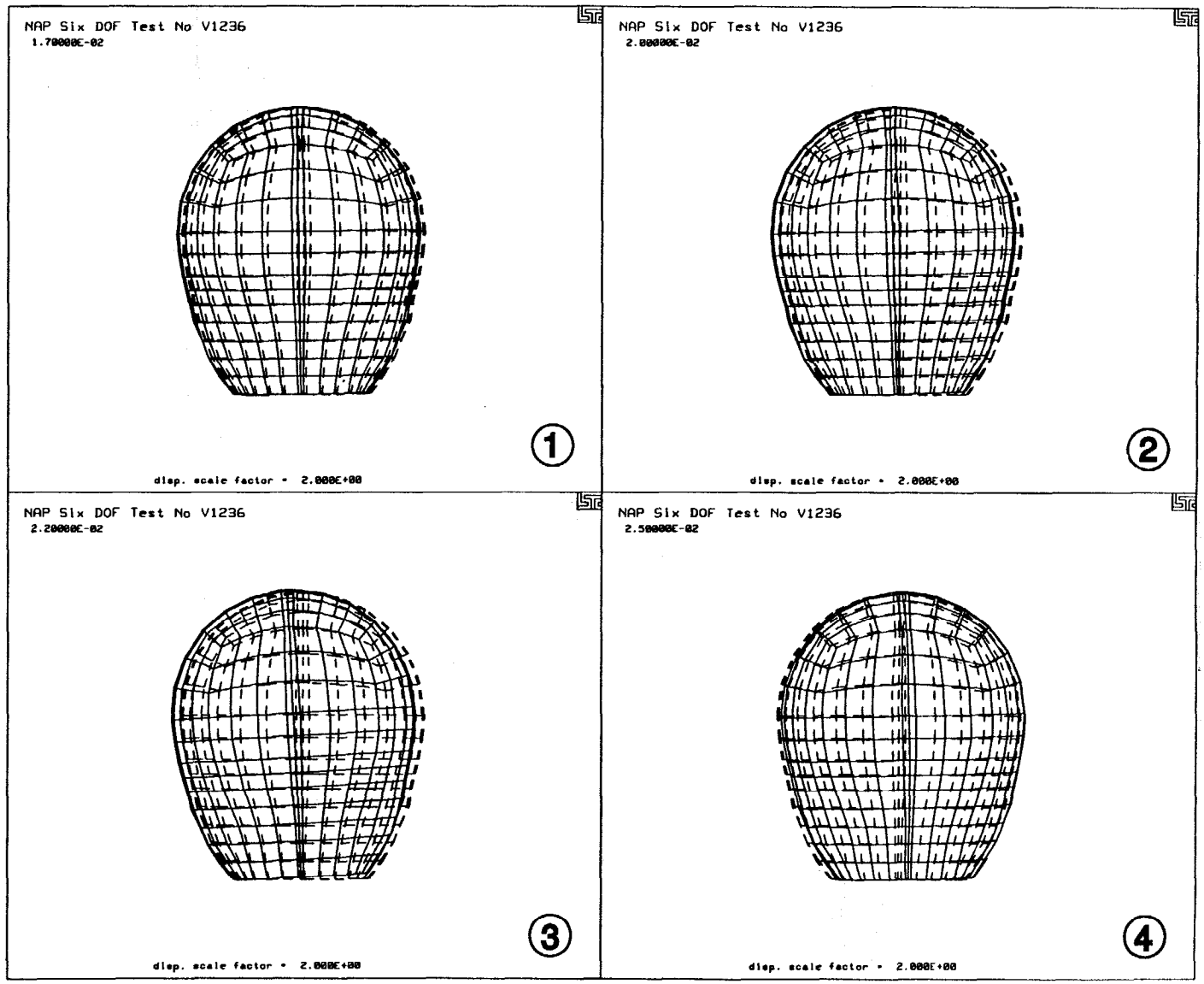


FIGURE 3-28. HEADPART ROLL MOTION FROM REAR VIEW RELATIVE TO INITIAL POSITION (17 - 25 MSEC)

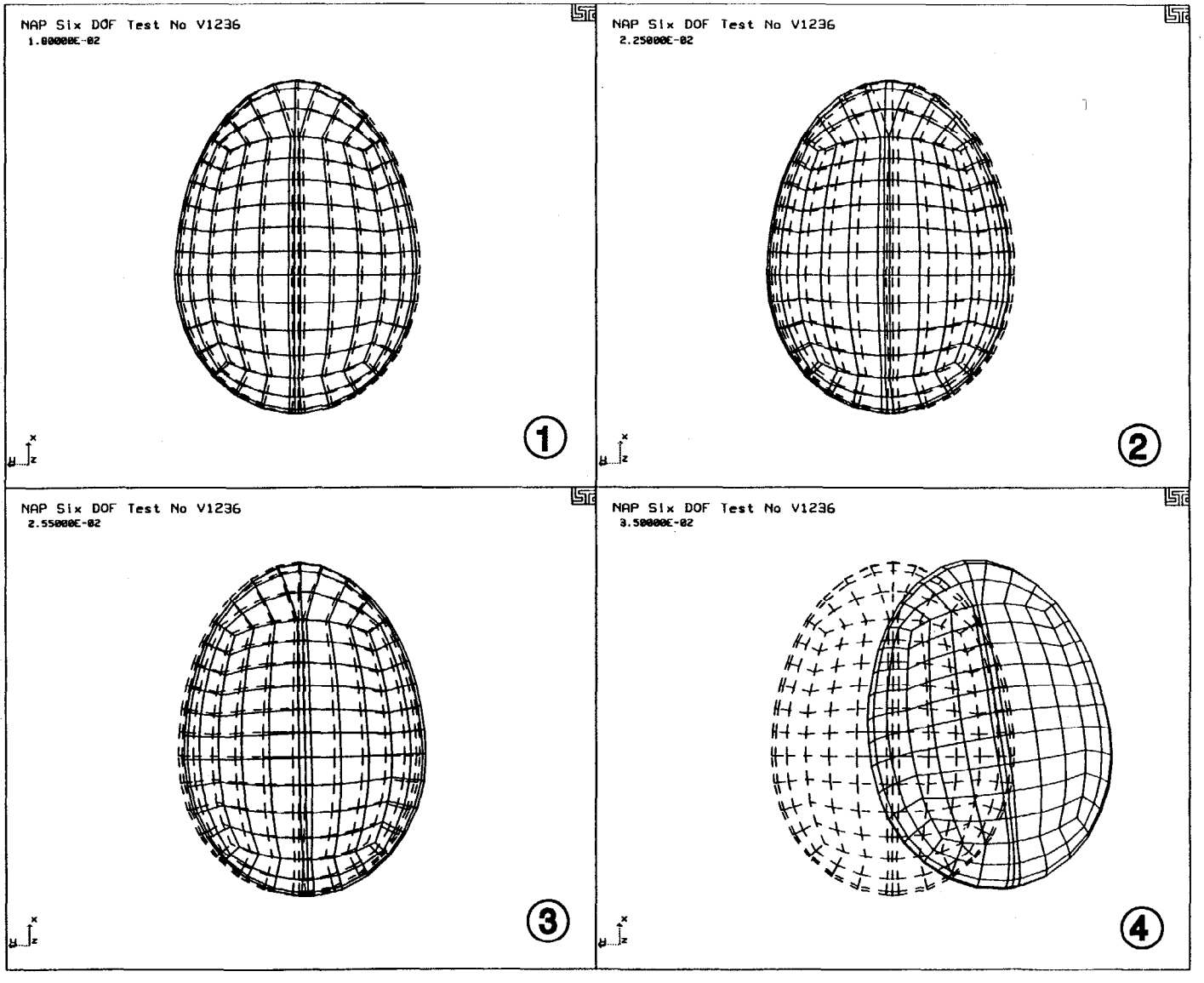


FIGURE 3-29. HEADPART YAW MOTION FROM PLAN VIEW RELATIVE TO INITIAL POSITION (10 - 35 MSEC)

Figure 3-28, the roll motion of the headpart towards the side structure is clear. In the fourth tile, at 25 msec, the headpart is starting to move away from the side structure. Similar data from a plan view is presented in Figure 3-29, without scaling, to illustrate kinematics associated with B-pillar impact. In the first two tiles, the head is approaching the B-pillar, and in the third and fourth tiles, a yaw motion of the headpart is clearly visible which implies head contact at a location posterior to the headpart CG.

The subtle kinematic responses illustrated in Figures 3-28 and 3-29 agree very well with expected impact mechanics for this scenario.

3.11 COMPARISON OF SIDE IMPACT KINEMATICS WITH FILM DATA

Figures 3-30 through 3-33 illustrate a more complete kinematic response sequence using selected frames from film data and the corresponding simulation. Similar to the previous example, Figures 3-30a and 3-30b, respectively illustrate the first half of a comparison of film data versus simulated headpart kinematics as viewed from a sled-mounted camera located directly behind the dummy head. Figures 3-31a and 3-31b illustrate the second half of this sequence. Since the film data for this view was taken from a location directly behind the dummy headpart, differences due to position and perspective should be minimal.

This sequence indicates very little motion initially, with the head subsequently rolling slightly towards the B-pillar (i.e., a negative angular rotation about the x body axis) presumably resulting from shoulder contact with the door, before a more complicated combination of pitch and yaw motion occurs after B-pillar impact. This was followed by large lateral displacements of the headpart across the occupant compartment after impact. While an exact characterization of headpart kinematics based on film data analysis would be very difficult to make (as well as introduce an additional uncertainty related to the analysis), it can be seen that the overall character of the impact kinematics is very well represented by the simulation.

A second view is provided in Figures 3-32a and 3-32b and Figures 3-33a and 3-33b, for a sled-mounted camera located opposite the dummy, and aligned nearly parallel to the sled axis. Comparing kinematics from this elevation view is more difficult because of significant differences in perspective and uncertainties in viewing distance, but provides a second view of the previously discussed kinematic response. Failed volume strains for this simulation are shown in Figure 3-34 for principal strain values of 5, 7, 9, and 11 percent and, again indicates a trend of increased accumulation of strain in the deformable material due to post-impact angular kinematics.

FIGURE 3-30A. FILM DATA FROM REAR MOUNTED CAMERA (FIRST SEQUENCE- SIDE IMPACT TEST)

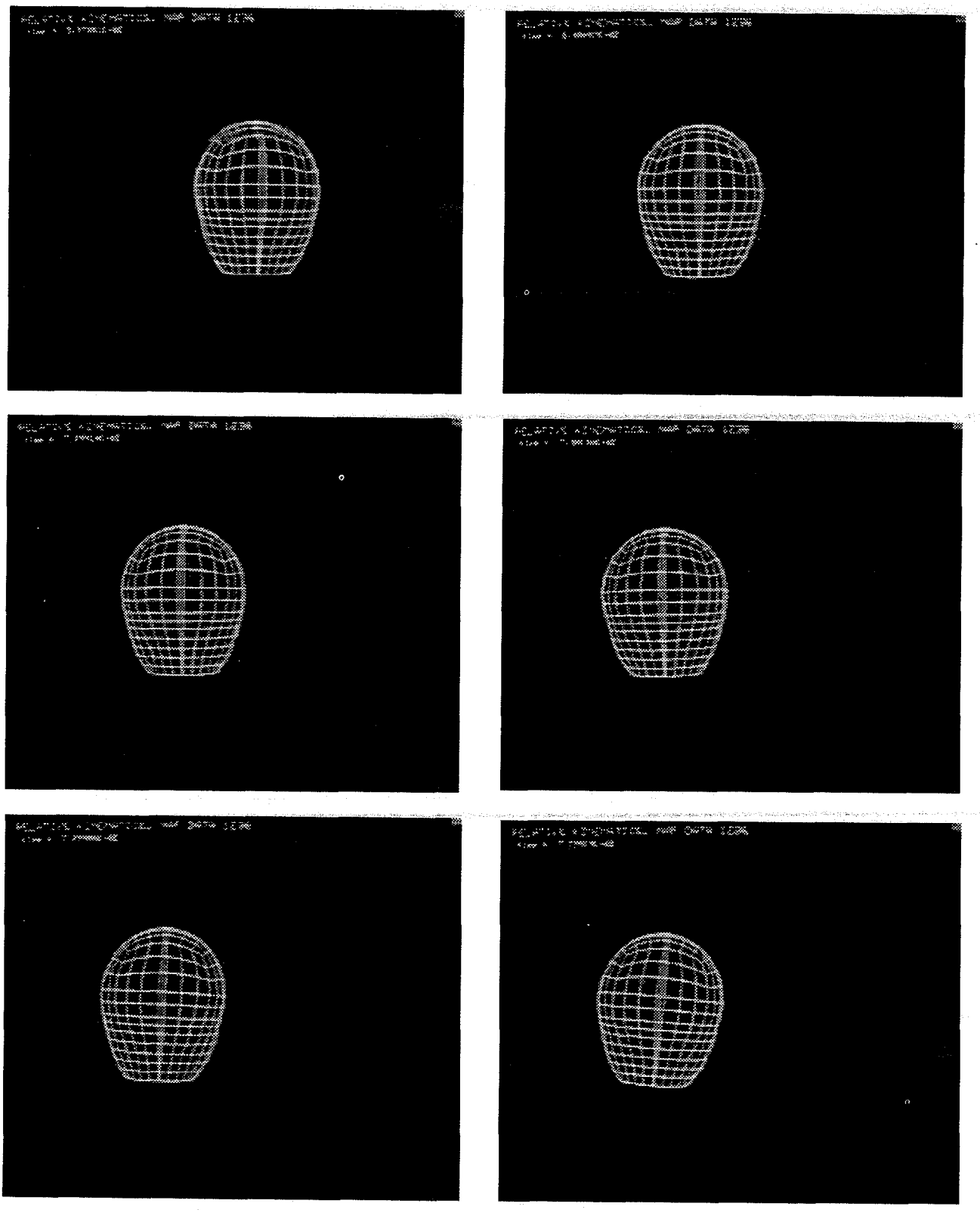


FIGURE 3-30B. SIMULATED KINEMATICS VIEWED FROM BEHIND (FIRST SEQUENCE - SIDE IMPACT TEST)

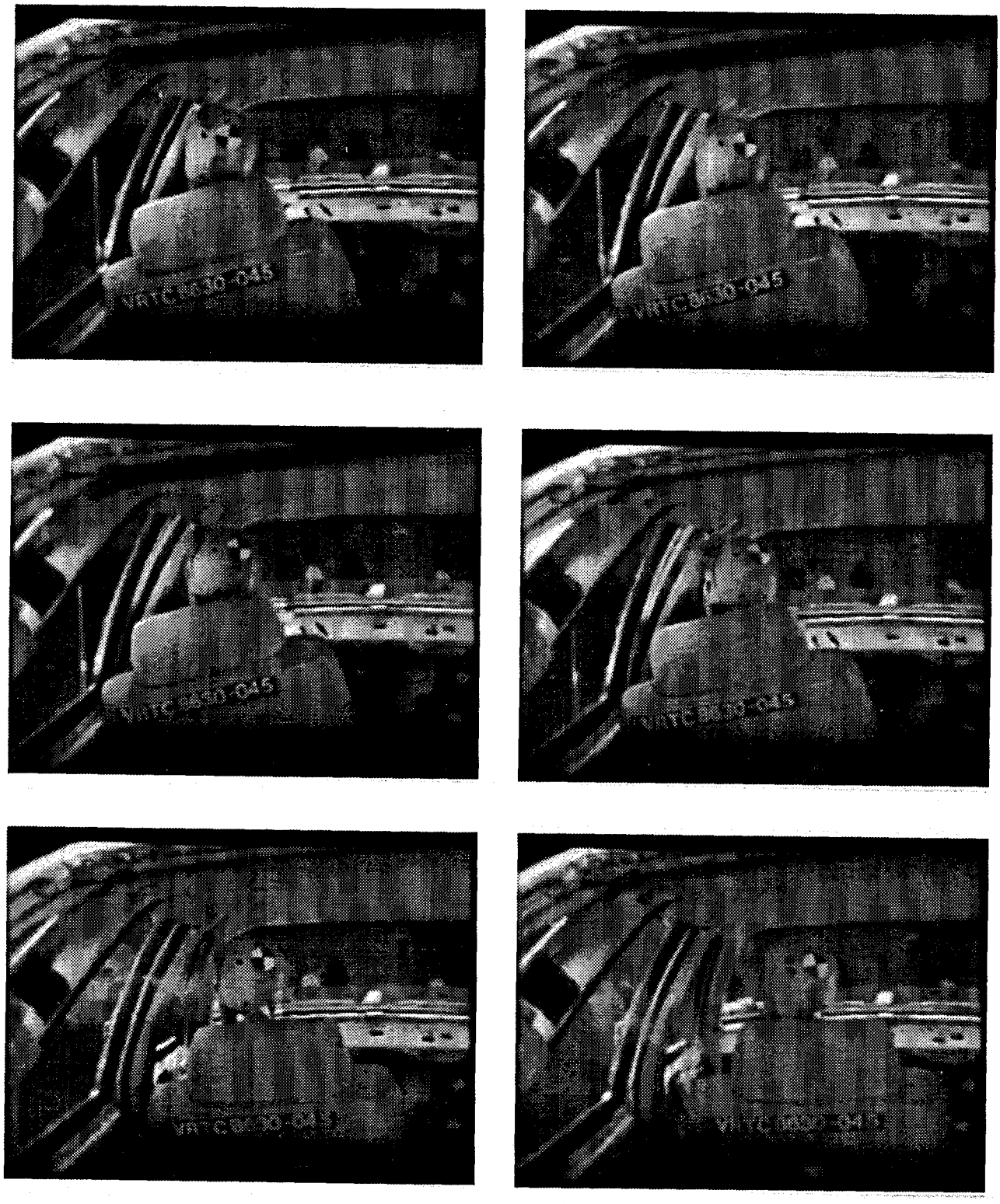


FIGURE 3-31A. FILM DATA FROM REAR MOUNTED CAMERA (SECOND SEQUENCE - SIDE IMPACT TEST)

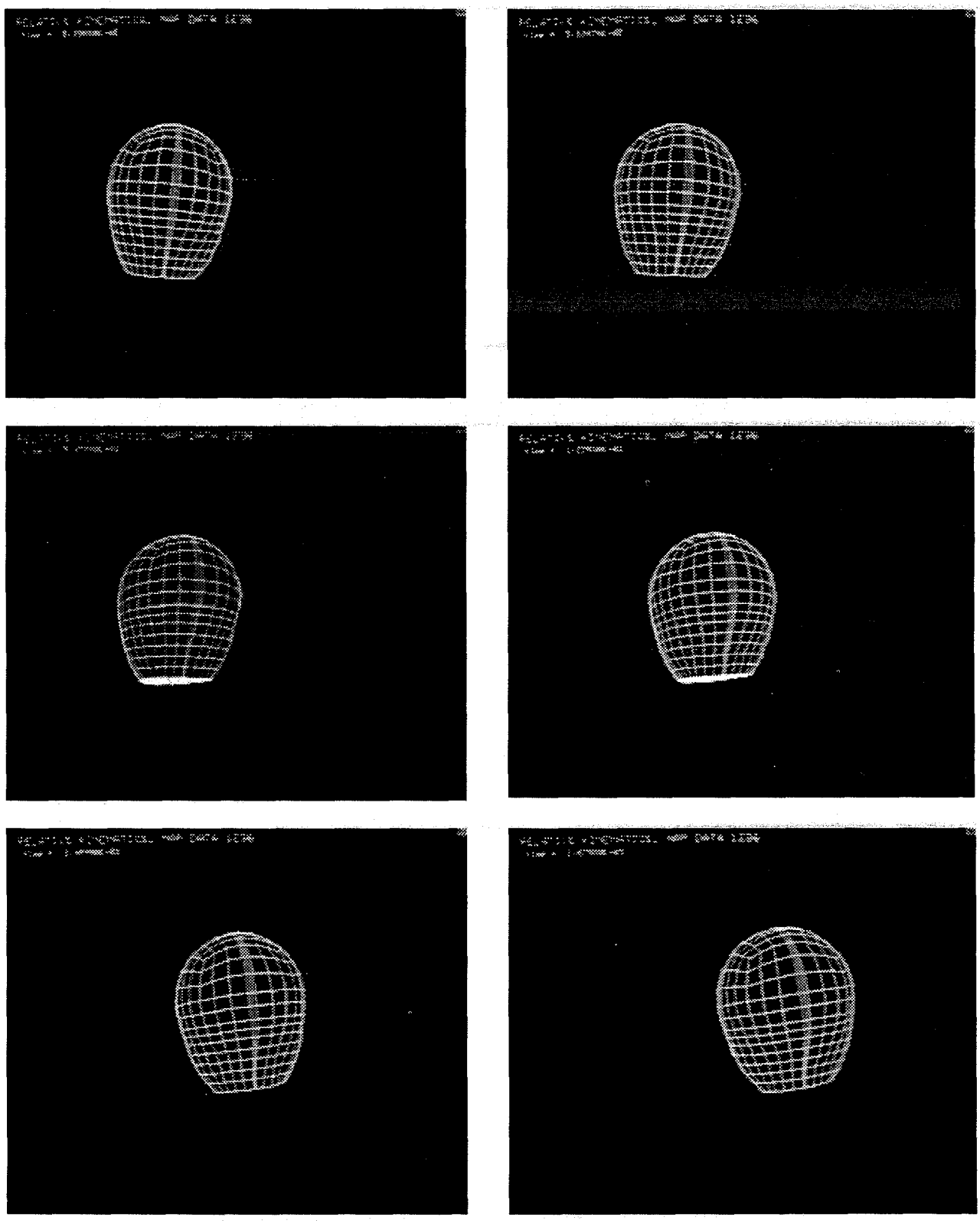


FIGURE 3-31B. SIMULATED KINEMATICS VIEWED FROM BEHIND (SECOND SEQUENCE - SIDE IMPACT TEST)

FIGURE 3-32A. FILM DATA FROM RIGHT SIDE MOUNTED CAMERA (FIRST SEQUENCE - SIDE IMPACT TEST)

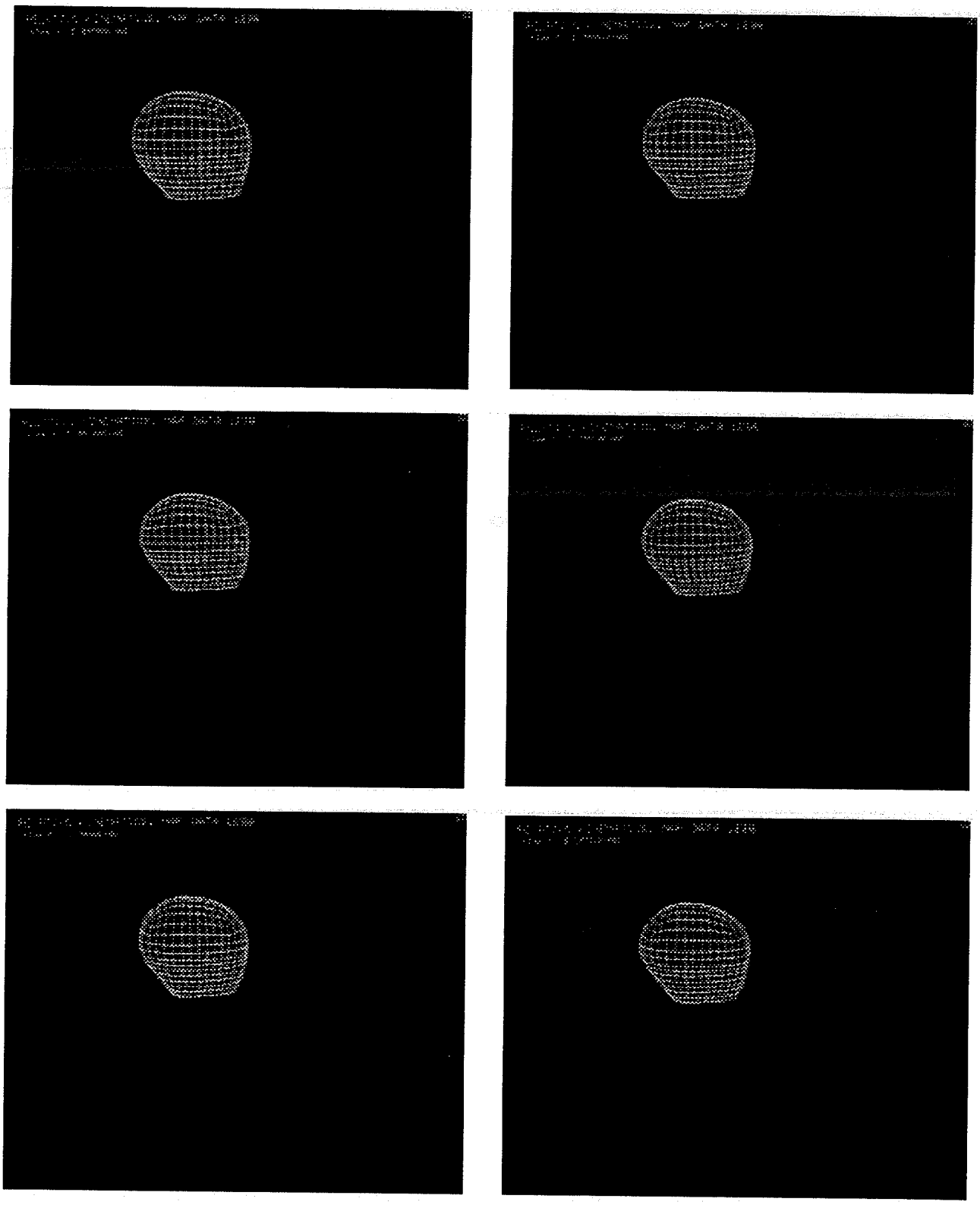


FIGURE 3-32B. SIMULATED KINEMATICS VIEWED FROM RIGHT SIDE (FIRST SEQUENCE - SIDE IMPACT TEST)

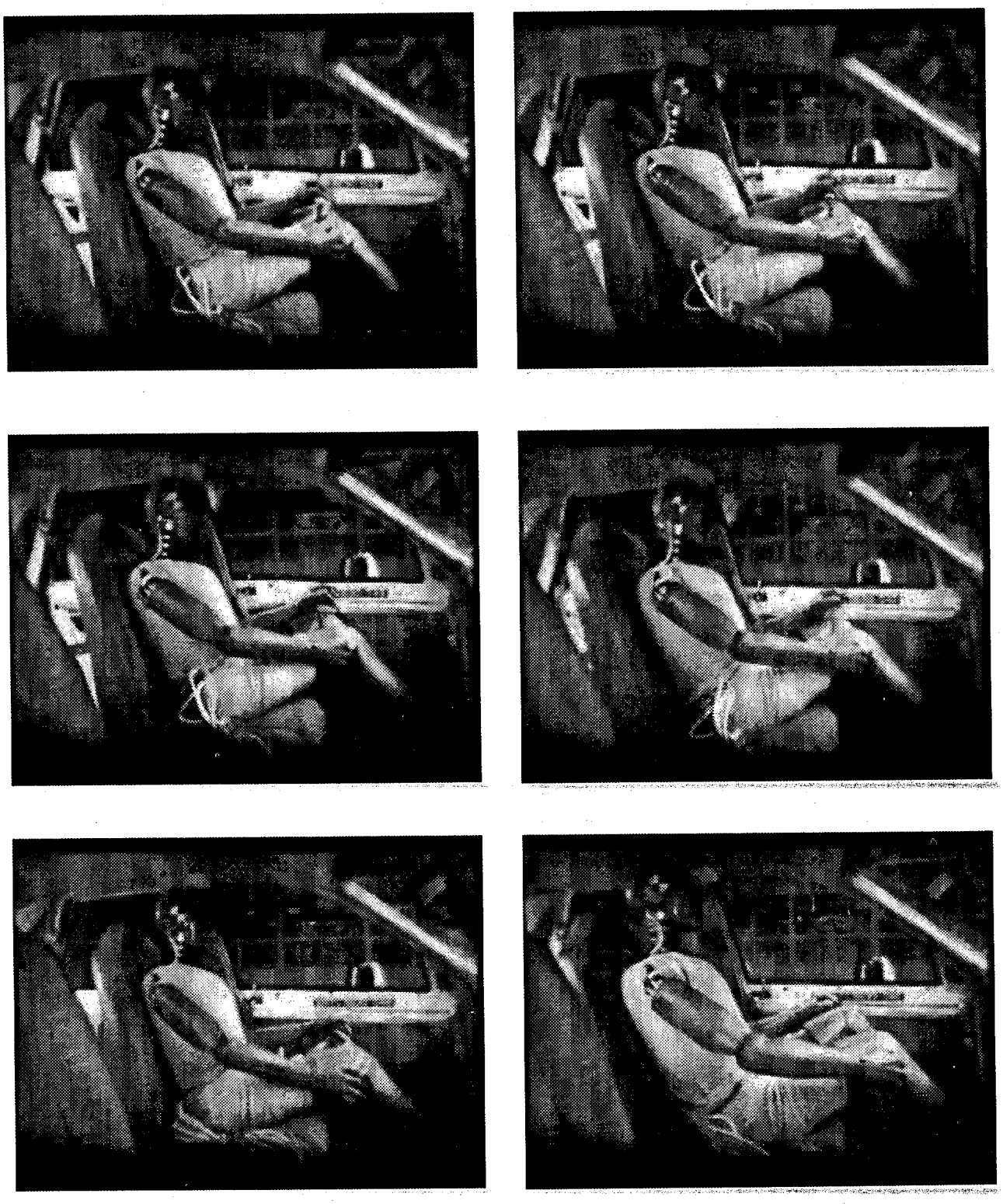


FIGURE 3-33A. FILM DATA FROM RIGHT SIDE MOUNTED CAMERA (SECOND SEQUENCE - SIDE IMPACT TEST)

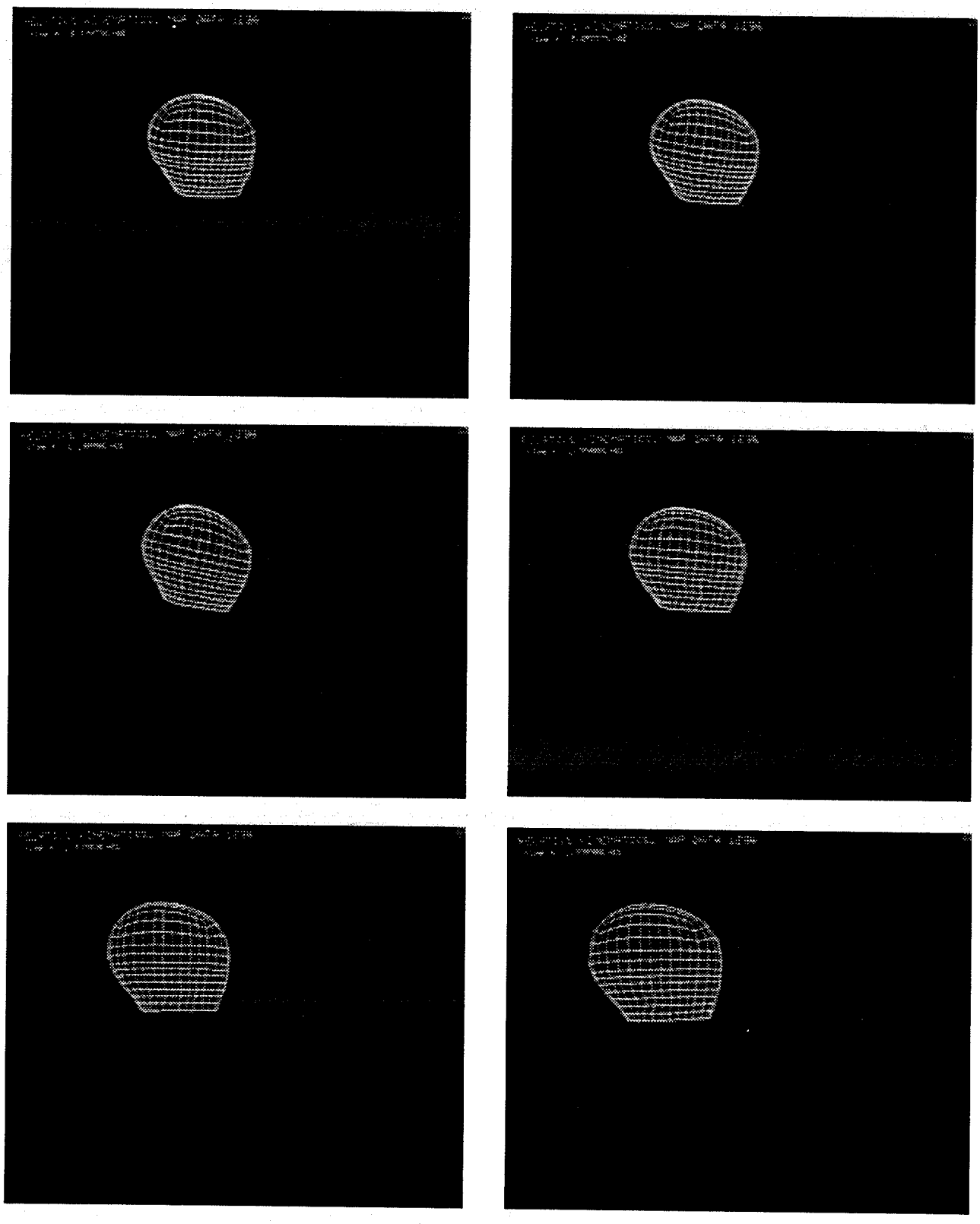


FIGURE 3-33B. SIMULATED KINEMATICS VIEWED FROM RIGHT SIDE (SECOND SEQUENCE - SIDE IMPACT TEST)

4. VERIFICATION OF TRANSFORMATION PROCESS USING SIMULATED NAP DATA

4.1 SOME DIFFICULTIES WITH FILM DATA

Verification of the transformation software based solely on comparison of test film data versus simulations using NAP data involves several difficulties, each with the potential to introduce some form of uncertainty. In the previously discussed comparisons, it was necessary to compute relative kinematics between headpart and vehicle (or sled in laboratory testing) and compare kinematics with film data from a vehicle-mounted camera to get a good view of headpart contact with interior surfaces. Because of the displacements typically associated with crash test data, providing a good comparison of simulated versus film data is very difficult because of uncertain viewing angles, viewing positions, and effects of perspective. Clear film footage and simple viewing angles are of great help, but these are often unavailable, especially when large multi-axis angular rotations are present. In addition, the measurement data itself must be acquired and recorded without introducing error into the system. (Although accelerometer data can be checked for obvious flaws, the absolute quality of the measurement data is very difficult to guarantee with absolute certainty.) The analysis of film data to quantify kinematic response can introduce yet another uncertainty in the form of errors introduced in the digitization or analysis process.

Using measurement data alone also leaves little opportunity to perform sensitivity studies to identify relative errors associated with the transformation process itself, versus geometric uncertainties such as non-coplanar geometry, and transducer errors such as cross-axis sensitivity. Finally, the comparison of a dummy headpart position in film versus simulation data will always involve uncertainties resulting from qualitative judgments and/or data reduction errors associated with film analysis. This is particularly so for a body with complicated geometry undergoing large generalized displacements.

4.2 USING SIMULATED NAP DATA FOR TESTING TRANSFORMATION SOFTWARE

In order to supplement comparisons with film data, simulated accelerometer data has been generated to simulate angular kinematics for two scenarios with known final angular positions. The first scenario involves *sequential* orthogonal rotations about the body x, y, and z axes, and the second scenario uses a rigid body rotation about a single fixed axis and a skewed angular orientation of the headpart to produce NAP data which results in *simultaneous* angular rotations about the body x, y, and z axes in the body coordinate system. For each of these conditions, the final position of the rotated headpart (or rotated object), is known or may be computed, and may be compared with the final position of

the object rotated with prescribed kinematics derived from simulated and transformed NAP data. Simulated accelerometer data may also be used to assess the relative influence of principal transducer error sources such as cross-axis sensitivity, and to assess the effect of non-coplanar geometry in the transformation process. Effects on variations from exact rotational kinematics on strain calculations may be made as well. After reviewing these issues for the latter scenario, an algorithm was developed to compensate for non-coplanar geometry effects, as described below.

4.3 SEQUENTIAL ROTATIONS ABOUT BODY x, y, and z AXES

For combinations of orthogonal rotations, the final position of a rotated object is both known and easily compared with the simulated result. One such scenario is the application of sequential orthogonal rotations to a rectangular object with distinctive dimensions along each orthogonal direction, about each body axis. The final position of the book-shaped hexahedral object is a function of the sequence of rotations, and has a distinctive final position which may be demonstrated by applying similar rotations to a physical object such as a textbook. The known final position is easily compared with the sequential rotations produced by simulated NAP data transformed from body to fixed coordinates, as illustrated below.

The following steps were taken to generate simulated NAP data for sequential rotations about the body x, y, and z axes:

A. Translational accelerations were computed at each location in the NAP array, using a haversine (1-cos [ωt]) pulse shape for defining angular displacements. Simulated NAP data was then used to affect sequential rotations about the body x, y, and z axes. Since this scenario involves pure rotation, accelerations located on the axis of rotation are zero, as are acceleration components normal to the plane of angular rotation.

B. The magnitude and sequence of translational accelerations were selected to provide the following combination of orthogonal rotations and null intervals:

Angular Rotation/ Null Interval	Body Axis	Interval (msec)	Time (msec)
90 degrees	x	40	0 - 40 msec
null interval	--	10	40 - 50 msec
90 degrees	y	40	50 - 90 msec
null interval	--	10	90 - 100 msec
90 degrees	z	40	100 - 140 msec

C. The simulated NAP data generated in body coordinates was transformed and used to generate load curves for rotating a "book-like" rigid object with the following dimensions along the body x, y, and z axes:

x axis dimension = 9.00 (length)
y axis dimension = 1.40 (thickness)
z axis dimension = 1.00 (width)

Load curves were integrated into a finite-element model of the book-like rigid object, and LS-DYNA3D was used for generating the kinematic results. (The finite-element representation of the rigid object is very simple to create, and computational resources are also minimal since the object being rotated is rigid and relatively large time steps may be used in the integration.)

D. The final position of the rotated object using the body-fixed translational accelerations was then compared to its known final orientation to quantify and illustrate the accuracy of the transformation process.

4.4 RESULTS - SEQUENTIAL ROTATIONS

Simulated accelerometer data was developed to produce pure sequential rotations of 90 degrees about the body-fixed x, y, and z axes. This combination of rotations is equivalent to sequential rotations of 90 degrees about the fixed X axis, followed by rotations about fixed Z and X axes (i.e., a second time). This can easily be verified by rotating an object of similar shape about stationary axes or axes attached to the object. Angular displacements of the form $(1-\cos[\omega t])$ were used to define angular displacement and corresponding translational acceleration pulses, to produce pure angular rotations of specified magnitude and duration.

A plot of the resulting kinetic energy time history, shown in Figure 4-1, serves to illustrate the temporal sequence of these rotations. The resulting angular kinematics are shown in Figures 4-2 through 4-4. Referring to the triad in the lower corner of each of these figures, it can be seen that these rotations are equivalent to rotations about the fixed X, Z and X axes, respectfully. This is reflected in the time histories for rigid body angular displacements about the fixed X, Y, and Z axes as shown in Figures 4-5 through 4-7, respectively. Corresponding angular velocity time histories about fixed axes are shown Figures 4-8 through 4-10.

Figure 4-11 illustrates an overplot of the object rotated via load curves generated from the NAP transformation package using simulated acceleration pulses versus a second object placed at the known final position by the previously mentioned sequence of rotations about the fixed coordinate axes (i.e., rx 90, rz 90, rx 90). It can be seen that the final

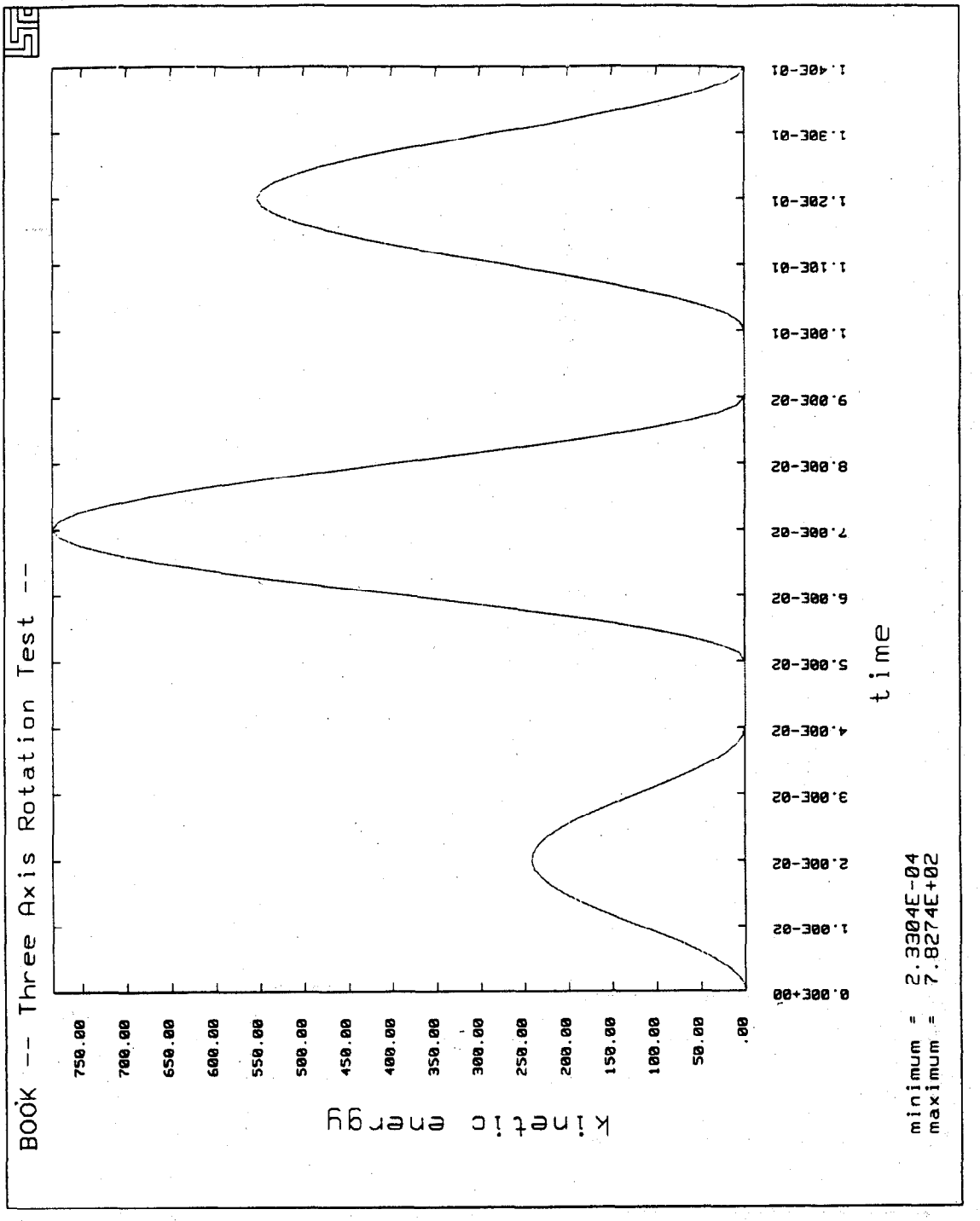


FIGURE 4-1. KINETIC ENERGY ASSOCIATED WITH ANGULAR ROTATIONS

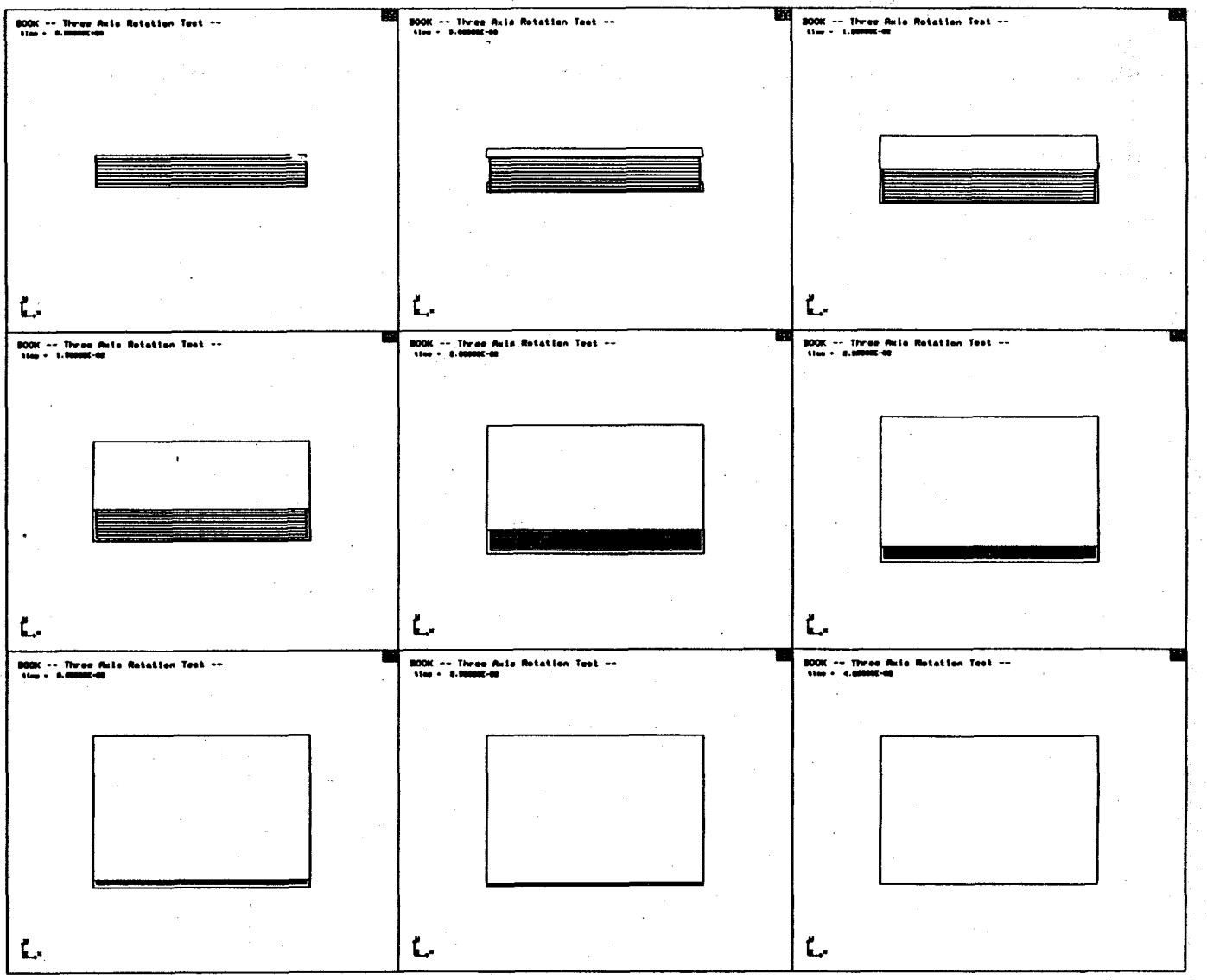


FIGURE 4-2. ANGULAR ROTATION SEQUENCE ABOUT X AXIS (0 - 40 MSEC)

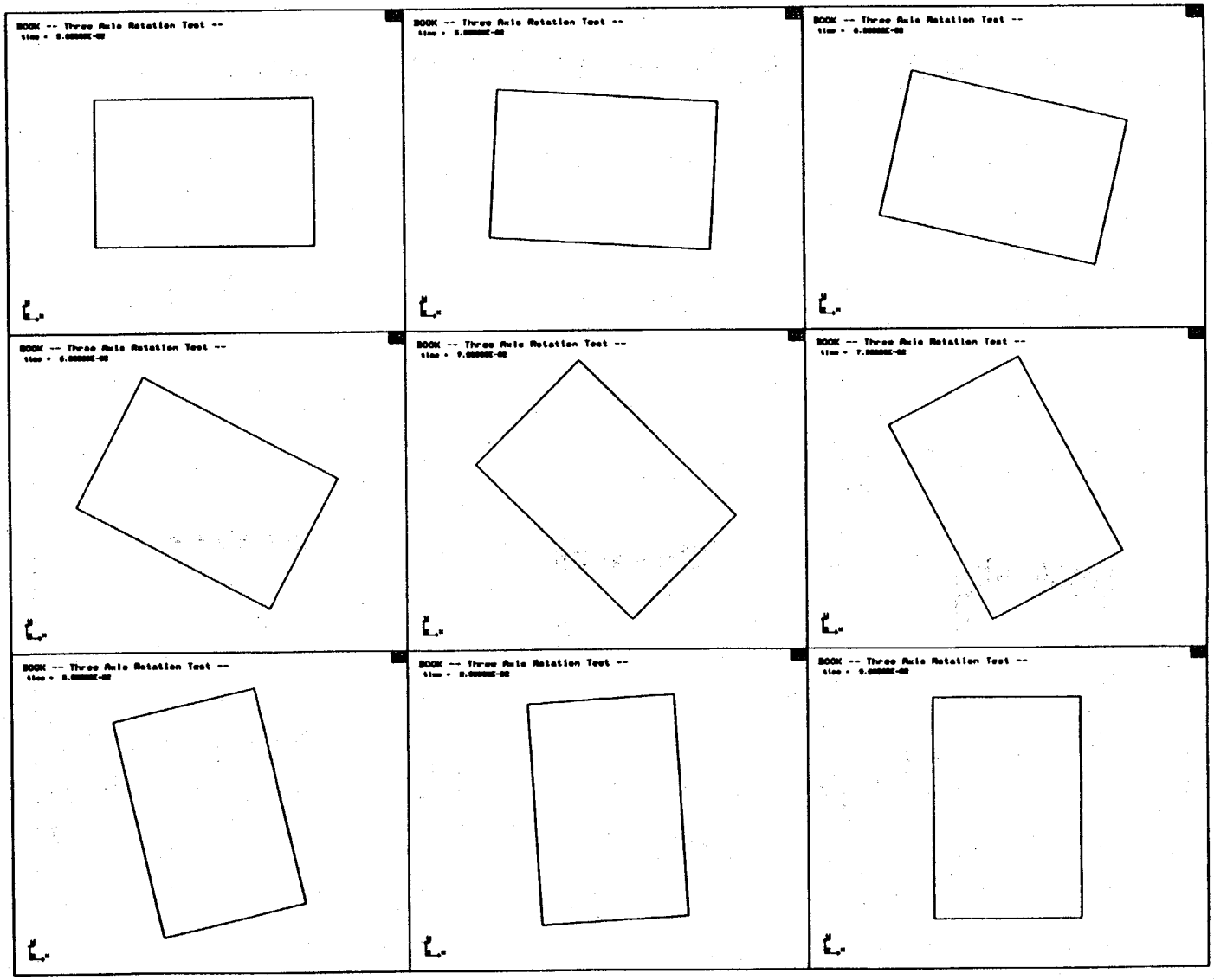


FIGURE 4-3. ANGULAR ROTATION SEQUENCE ABOUT Y AXIS (50 - 90 MSEC)

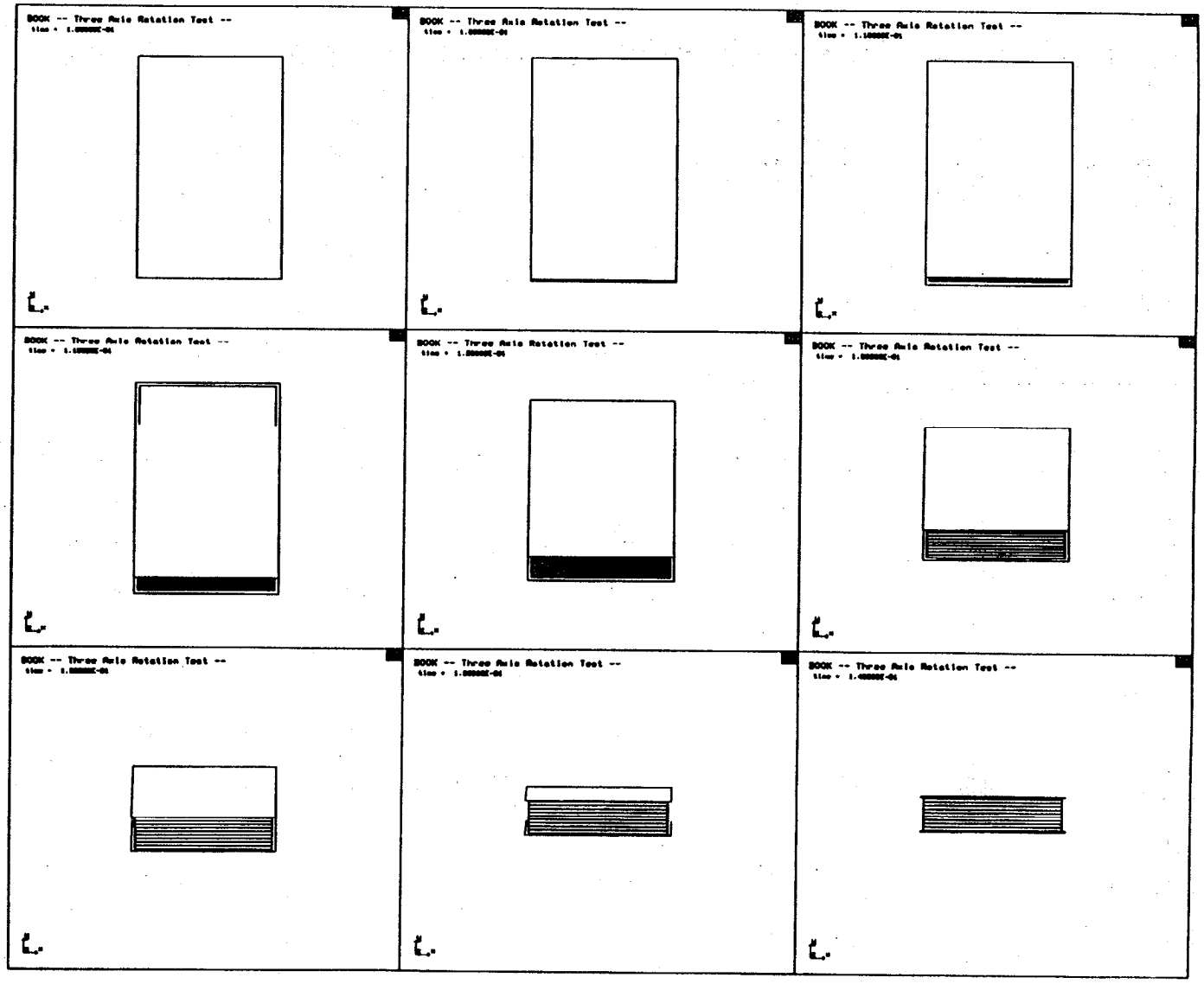


FIGURE 4-4. ANGULAR ROTATION SEQUENCE ABOUT Z AXIS (100 - 140 MSEC)

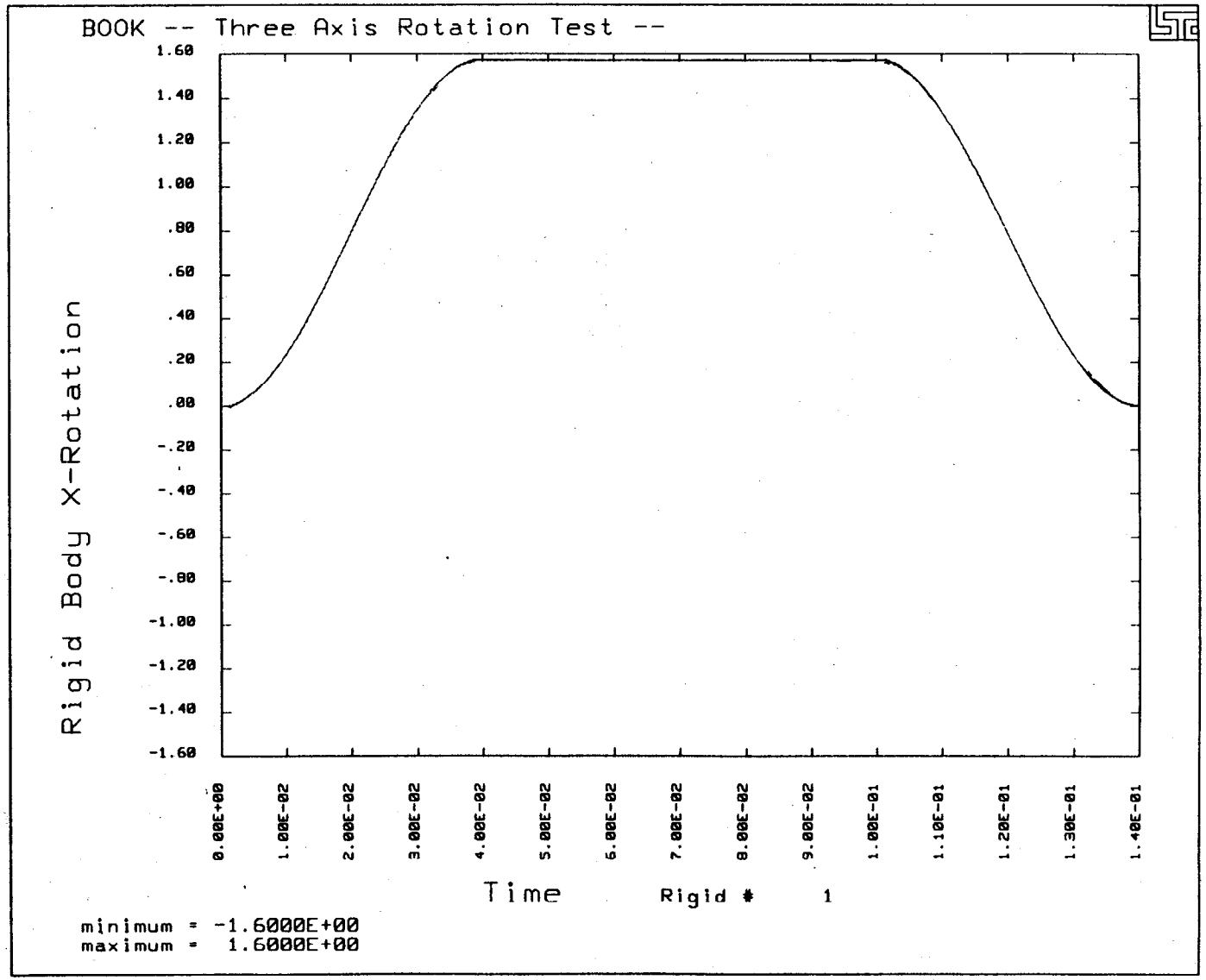


FIGURE 4-5. ANGULAR ROTATION ABOUT FIXED X AXIS VS TIME

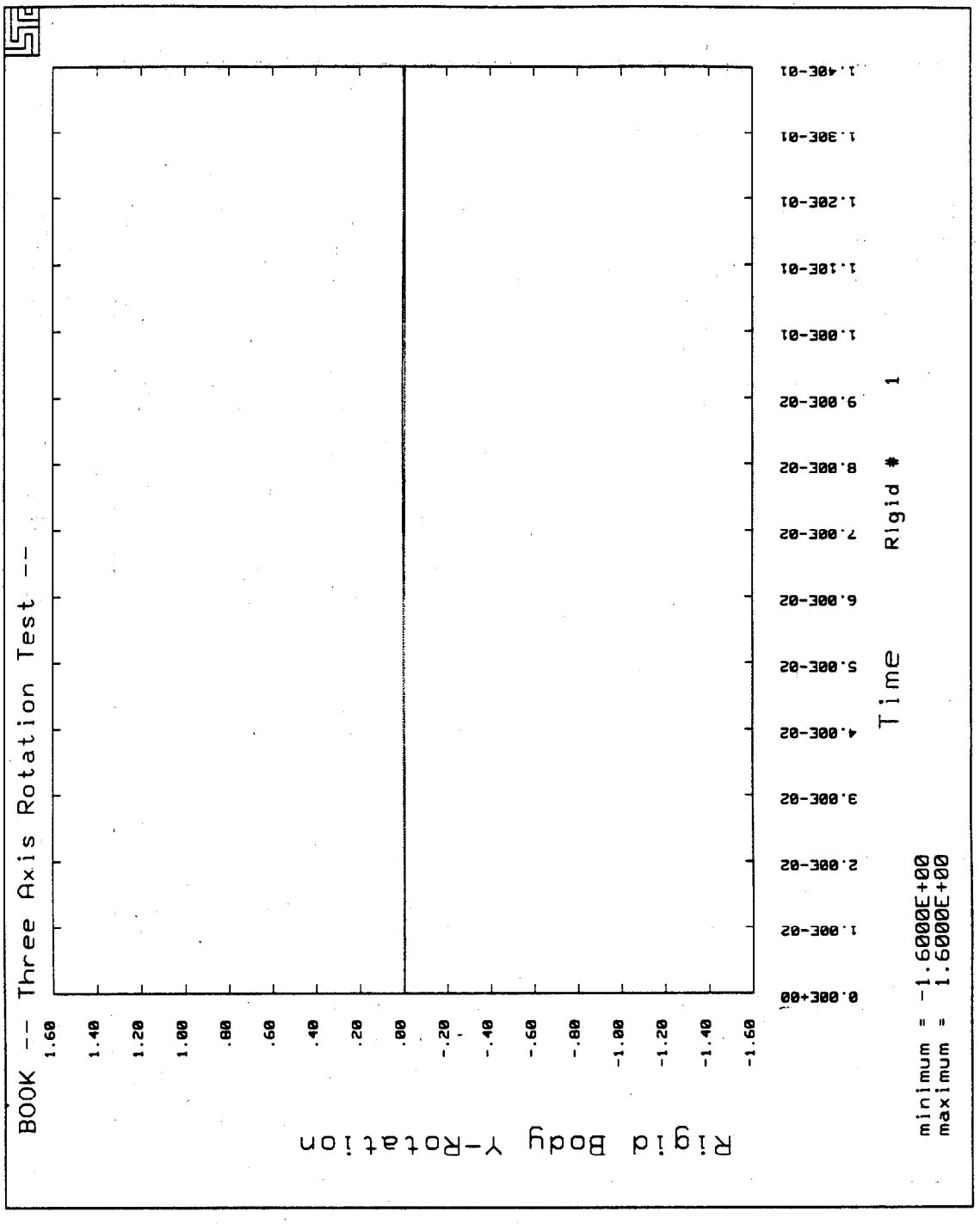


FIGURE 4-6. ANGULAR ROTATION ABOUT FIXED Y AXIS VS TIME

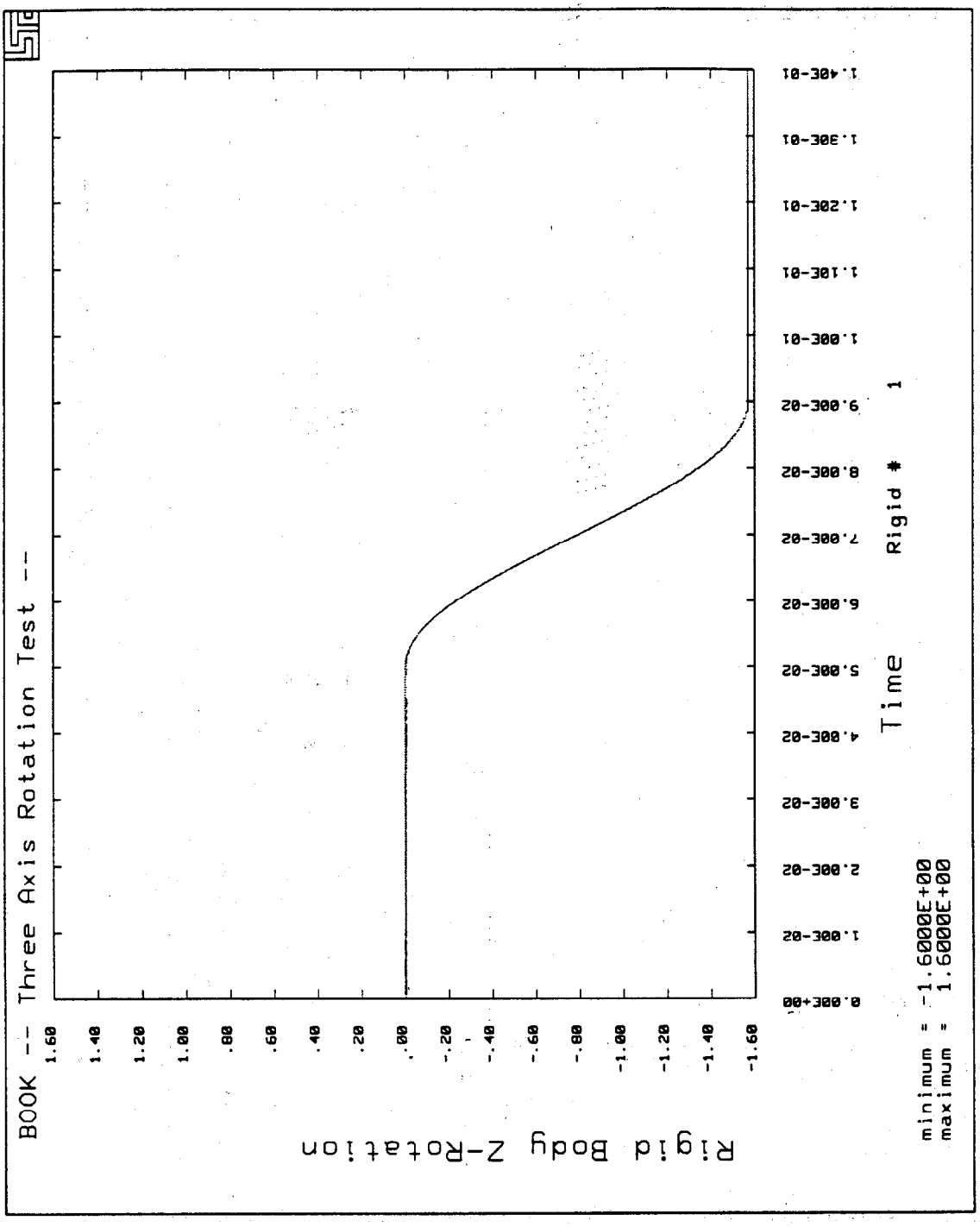


FIGURE 4-7. ANGULAR ROTATION ABOUT FIXED Z AXIS VS TIME

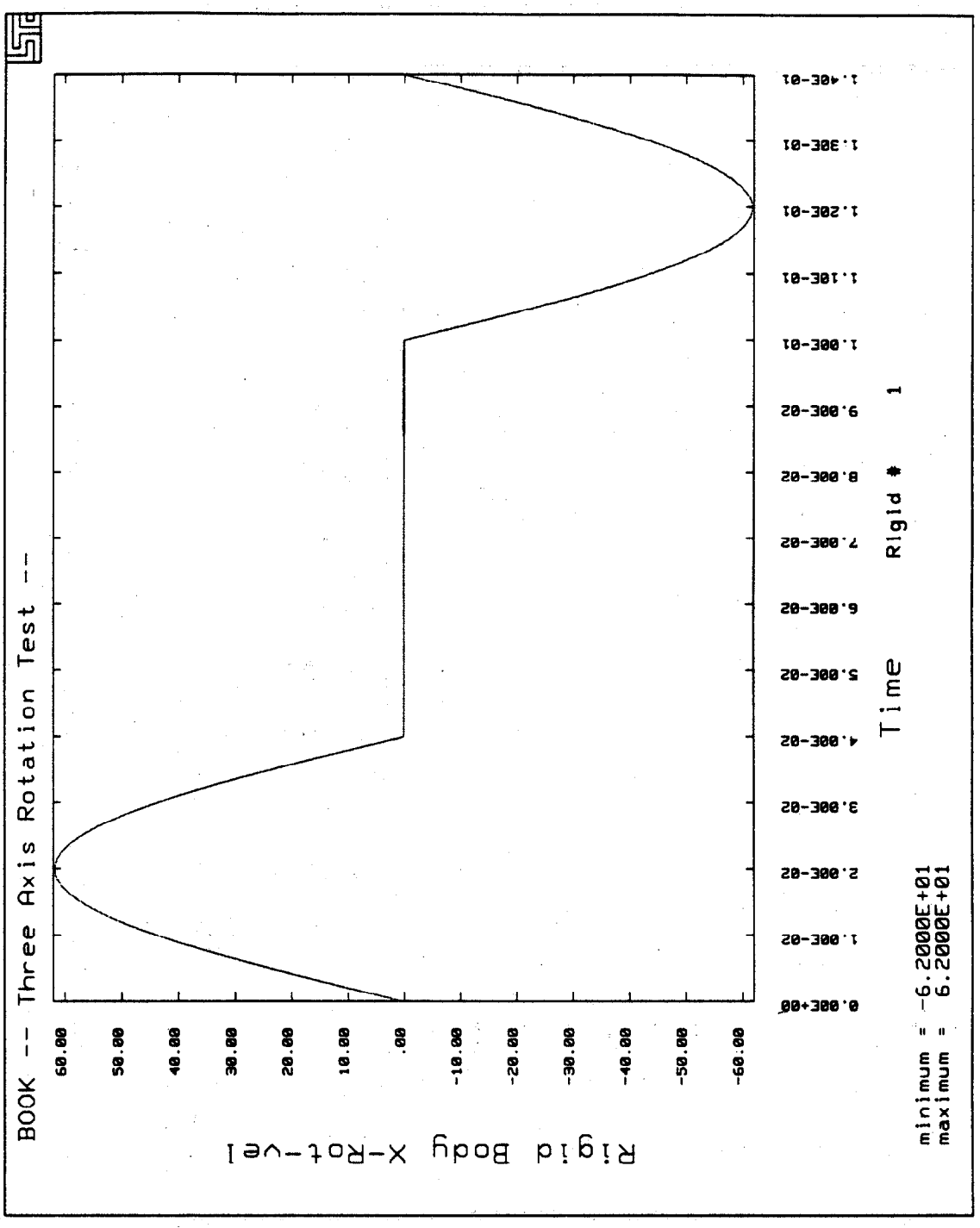


FIGURE 4-8. ANGULAR VELOCITY ABOUT FIXED X AXIS VS TIME

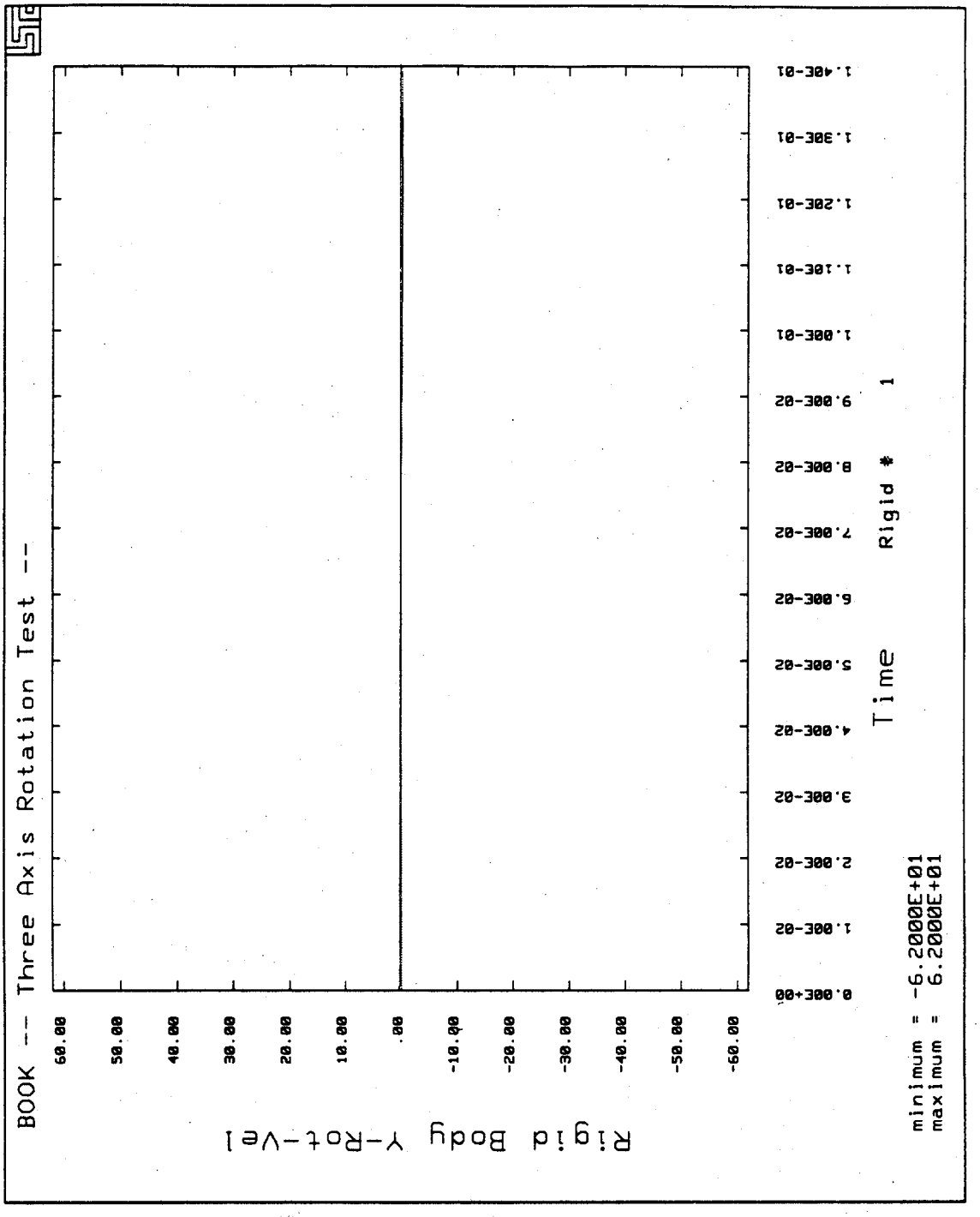


FIGURE 4-9. ANGULAR VELOCITY ABOUT FIXED Y AXIS VS TIME

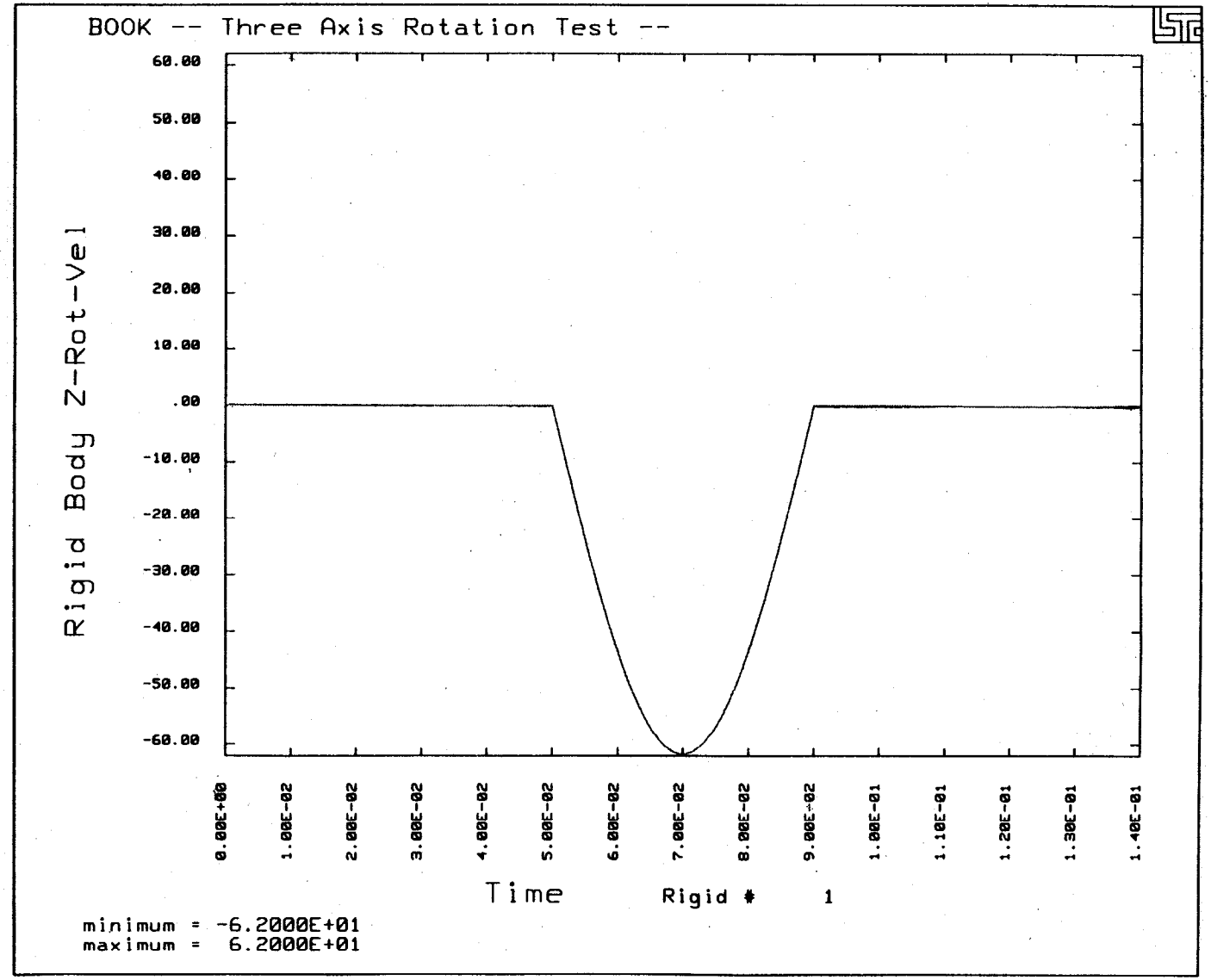


FIGURE 4-10. ANGULAR VELOCITY ABOUT FIXED Z AXIS VS TIME

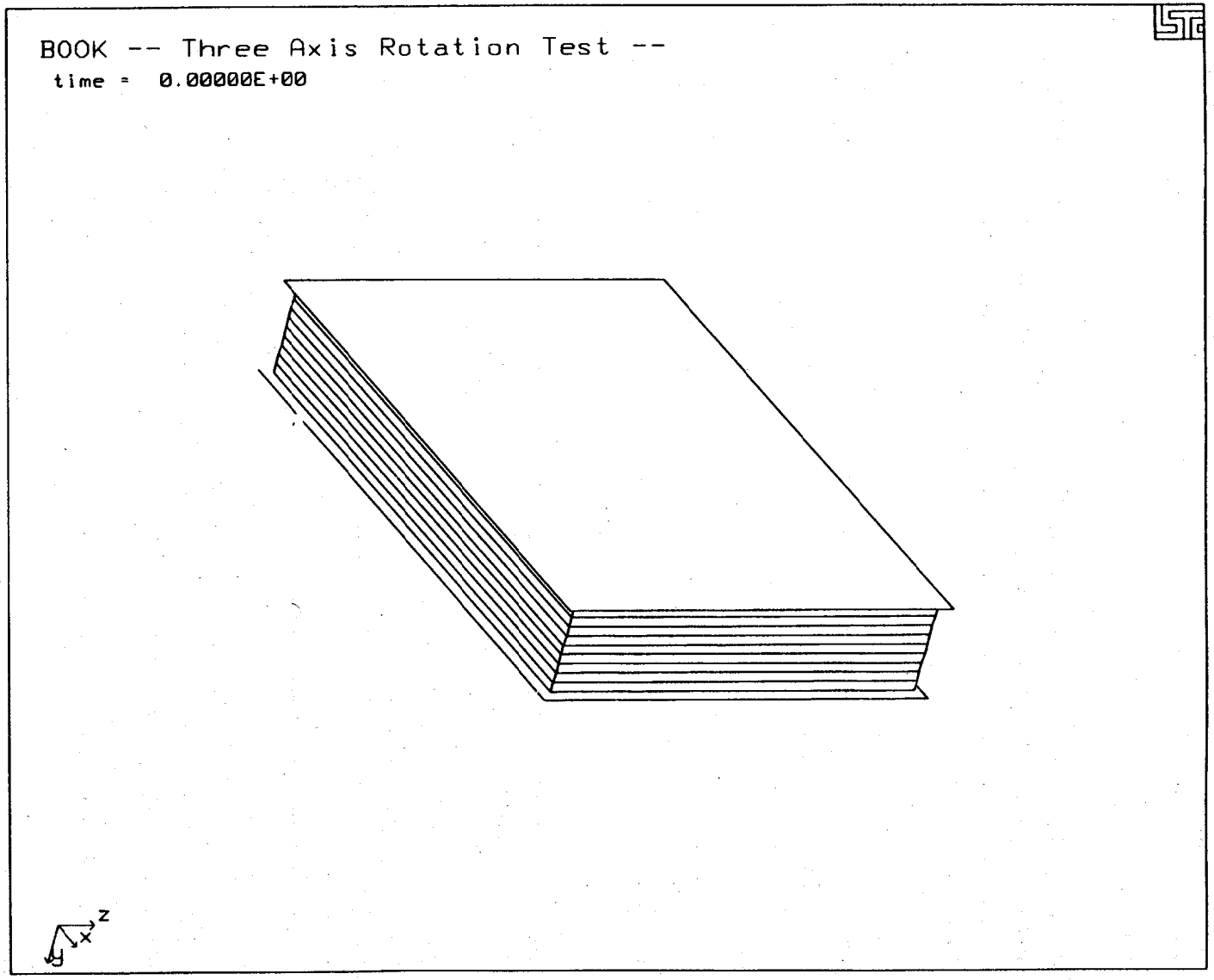


FIGURE 4-11. OVERLAY OF NAP SIMULATION VS CORRECT FINAL POSITION (OBLIQUE VIEW)

position of the rigid object rotated using the simulated NAP data (in terms of load curves prescribing translational and rotational velocities), and a second object placed at the final position, are essentially identical. These results indicate that the transformation process is effective at simulating sequential multi-axis rotations, and accurately replicates the applied angular kinematics.

4.5 SIMULTANEOUS ROTATIONS ABOUT BODY x, y, and z AXES

To provide a more challenging problem to check the accuracy of NAP transformation software and to evaluate relative uncertainties associated with transducer signals and non-coplanar geometry, simulated accelerometer data has been developed for the following condition:

A 60 msec haversine pulse was used to prescribe the angular velocity of a rigid headpart rotating about a fixed lateral axis. Pulse features were selected to generate a relatively large angular displacement and to provide a relatively long integration interval to maximize potential transformation errors as well as variations in deformable material strains which are sensitive to angular kinematics. To generate simultaneous angular velocity components about each of the three body axes for a single angular excitation applied about the fixed lateral axis, the rigid body was sequentially rotated about the body's principal x, y, and z axes to compute accelerometer data in the NAP array. This results in a skewed angular orientation with respect to the inertial axes.

Accelerometer data is generated at spatial positions corresponding to each transducer location within the NAP array, using both ideal and actual array geometry for comparison. Since the analytic pulse data provides accelerometer data without errors typically associated with measurement error, *the ideal geometry condition and computed NAP data provides a pure test of the transformation process, without introducing uncertainties associated with non-ideal geometry or errors associated with the measurement process.*

For perfect, or coplanar location of the accelerometers (i.e., sense axes of multiple accelerometer co-located at a single point within the array), the rotational displacement and final position of the headpart can be computed exactly and used to compare with results obtained using the transformation process. Variations from the known final position can then be evaluated in terms of effects due to non-coplanar geometry and/or transducer errors, as well as the evaluation of effects on strains induced in soft interior components.

4.5.1 Development of Simulated NAP Accelerometer Data

A brief description of the procedure utilized to develop simulated acceleration data for the condition of a rigid body rotation about a fixed lateral axis, and with the rigid body rotated about its x, y, and z axes, is outlined below. Figure 4-12 illustrates a rigid headpart, sequentially rotated by 0.5 radians or about 28 degrees, about the body x, y, and z axes. For the intended rigid body rotation about the lateral axis, the resulting motion should be restricted to the X-Z plane. The transformation matrix relating principal axes in the body-to-fixed coordinate systems, for sequential rotations about body x, y, and z axes (phi, theta, and psi) is determined from:

$$\begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}$$

The above transformation matrix, a radial offset distance between the center of gravity and the axis of rotation, and either idealized or non-coplanar accelerometer positions in the NAP array were used to compute acceleration time histories at each accelerometer position, using the following procedure.

Outline of Procedure to Develop Simulated NAP Data:

1. With rigid body axes originally parallel to the inertial reference frame, new transducer locations were computed after applying the sequential body rotations phi, theta, and psi, relative to the fixed lateral axis of rotation using the above transformation matrix. An initial radial offset distance of 10 inches was used between the headpart CG and the axis of rotation. This results in a new radial distance from the center of gravity to each accelerometer position in the rotated array.

2. A haversine angular velocity pulse was used to prescribe the rigid body angular rotation about the fixed Y, or lateral axis, and radial and tangential acceleration components were computed at each accelerometer position relative to the fixed coordinate system. The angular displacement is determined by integrating the angular velocity pulse, hence this motion results in a known angular rotation of the body in the X-Z plane.

3. Radial and tangential accelerations at each accelerometer location are resolved into the local body coordinate system, and the components along each accelerometer axis

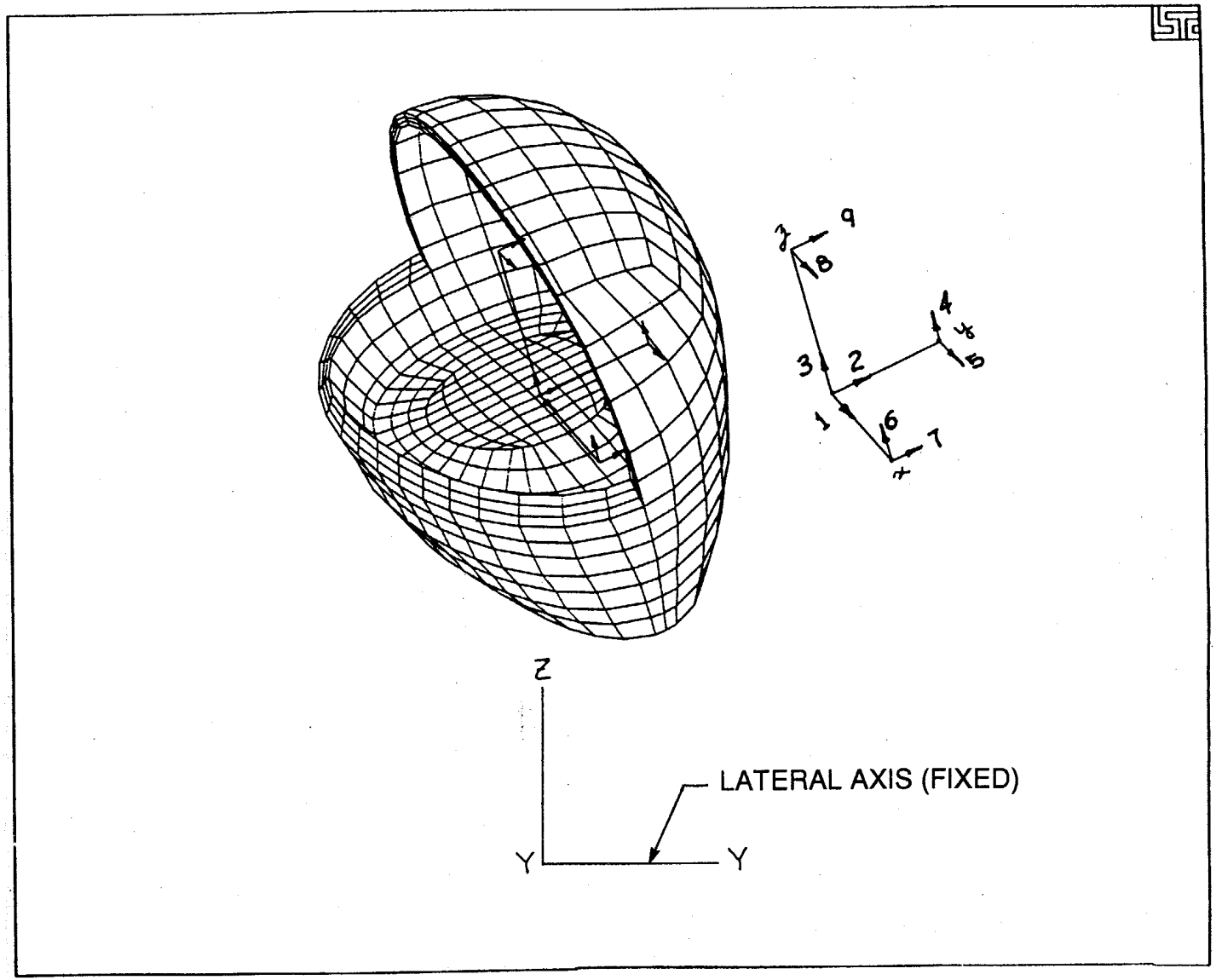


FIGURE 4-12. TRANSDUCER LOCATION (IDEAL GEOMETRY) - NAP ARRAY IN SECTIONED AND ROTATED HEADPART

are summed to compute the acceleration which would be measured by an accelerometer located at that point in the array.

4. Adjustments were made to the NAP transformation software to reflect the new initial relationship between body and fixed coordinate system using the inverse of the direction cosine matrix used to relate accelerations in the fixed coordinate system to the body coordinate system. (For a direction cosine matrix, the inverse of the matrix is equal to the transform.)

5. The nine acceleration time histories at each location were then computed as a function of: haversine pulse amplitude and duration; initial body rotations and either idealized or non-coplanar location of the accelerometers in the array. Tables 4-1 and 4-2 illustrate the geometry data used for ideal and non-coplanar geometry. The geometry of Table 4-2 corresponds to a general non-coplanar arrangement of accelerometers in a 3-2-2-2 NAP array used to illustrate the compensation algorithm and do not necessarily correspond to a specific design. Distances are from headpart x y and z axes passing through the cg to seismic centers of accelerometers.

6. Simulated pulses derived in this manner were then input to the NAP transformation software with the body rotation data, and load curve data was generated, using either ideal or non-coplanar geometry, to drive the rigid headpart.

7. A second headpart was positioned at the final reference position for comparison with results obtained from steps 1 through 6 above.

4.5.2 Haversine Pulse Features

A haversine angular velocity pulse shape was used to generate simulated accelerometer data. This results in a bi-modal resultant acceleration profile. A relatively long acceleration pulse was selected to give potential errors due to the integration process a chance to accumulate and to produce relatively large angular displacements which should accentuate differences in failed volume calculations. It was also desirable to select pulse features which resulted in a relatively high value for HIC.

For these reasons, an amplitude of 25 rad/sec and a duration of 60 msec was selected. This results in a peak angular velocity of 50 rad/sec. The resultant acceleration time history for this pulse is shown in Figure 4-13, and produces a HIC value of 1583 with $(t_2 - t_1)$= 54.3 - 5.73 = 48.54 msec. The angular acceleration is shown in Figure 4-14. Considering the moment arm of 10 inches from the center of rotation to the CG, the peak angular acceleration associated with this pulse is approximately 2500 rad/sec^2. The resulting angular displacement about the fixed lateral axis, is 85.94 degrees.

TABLE 4-1. POSITION OF ACCELEROMETERS IN NINE ACCELEROMETER PACKAGE (NAP) ARRAY

Idealized Geometry

Accel Location	Sense Axis	Accel Label	X axis Location	Y axis Location	Z axis Location
CG	X	Bacc1	0.000	0.000	0.000
CG	Y	Bacc2	0.000	0.000	0.000
CG	Z	Bacc3	0.000	0.000	0.000
Y Arm	Z	Bacc4	0.000	1.900	0.000
Y Arm	X	Bacc5	0.000	1.900	0.000
X Arm	Z	Bacc6	2.200	0.000	0.000
X Arm	Y	Bacc7	2.200	0.000	0.000
Z Arm	X	Bacc8	0.000	0.000	3.200
Z Arm	Y	Bacc9	0.000	0.000	3.200

TABLE 4-2. POSITION OF ACCELEROMETERS IN NINE ACCELEROMETER PACKAGE (NAP) ARRAY

General Non-Coplanar Geometry

Accel Location	Sense Axis	Accel Label	X axis Location	Y axis Location	Z axis Location
CG	X	Bacc1	0.203	0.000	0.000
CG	Y	Bacc2	0.000	0.203	0.000
CG	Z	Bacc3	0.000	0.000	0.203
Y Arm	Z	Bacc4	0.000	2.103	0.000
Y Arm	X	Bacc5	0.000	1.900	0.203
X Arm	Z	Bacc6	2.403	0.000	0.000
X Arm	Y	Bacc7	2.200	0.000	0.203
Z Arm	X	Bacc8	0.203	0.000	3.200
Z Arm	Y	Bacc9	0.000	0.203	3.200

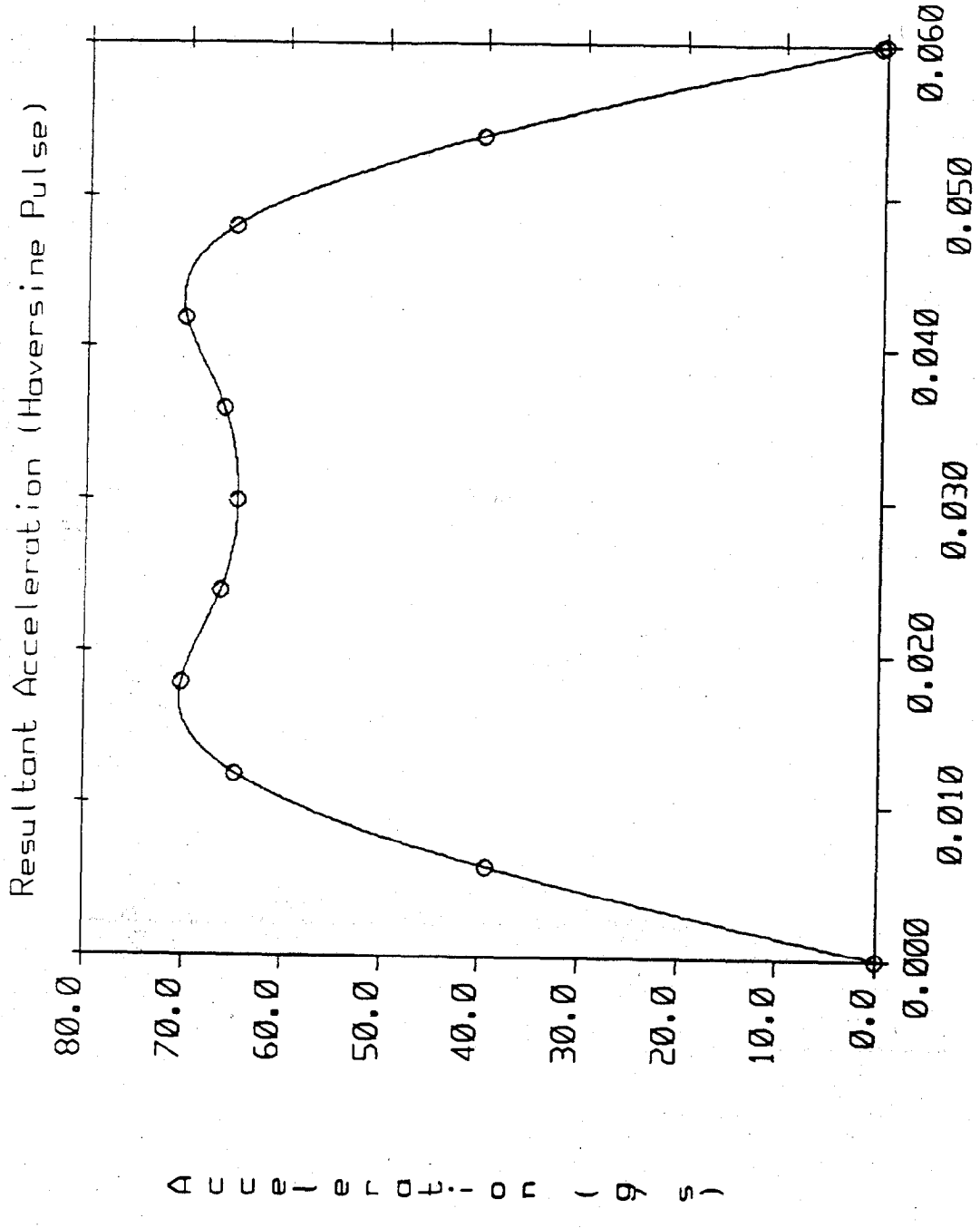


FIGURE 4-13. RESULTANT ACCELERATION TIME HISTORY HAVERSINE PULSE

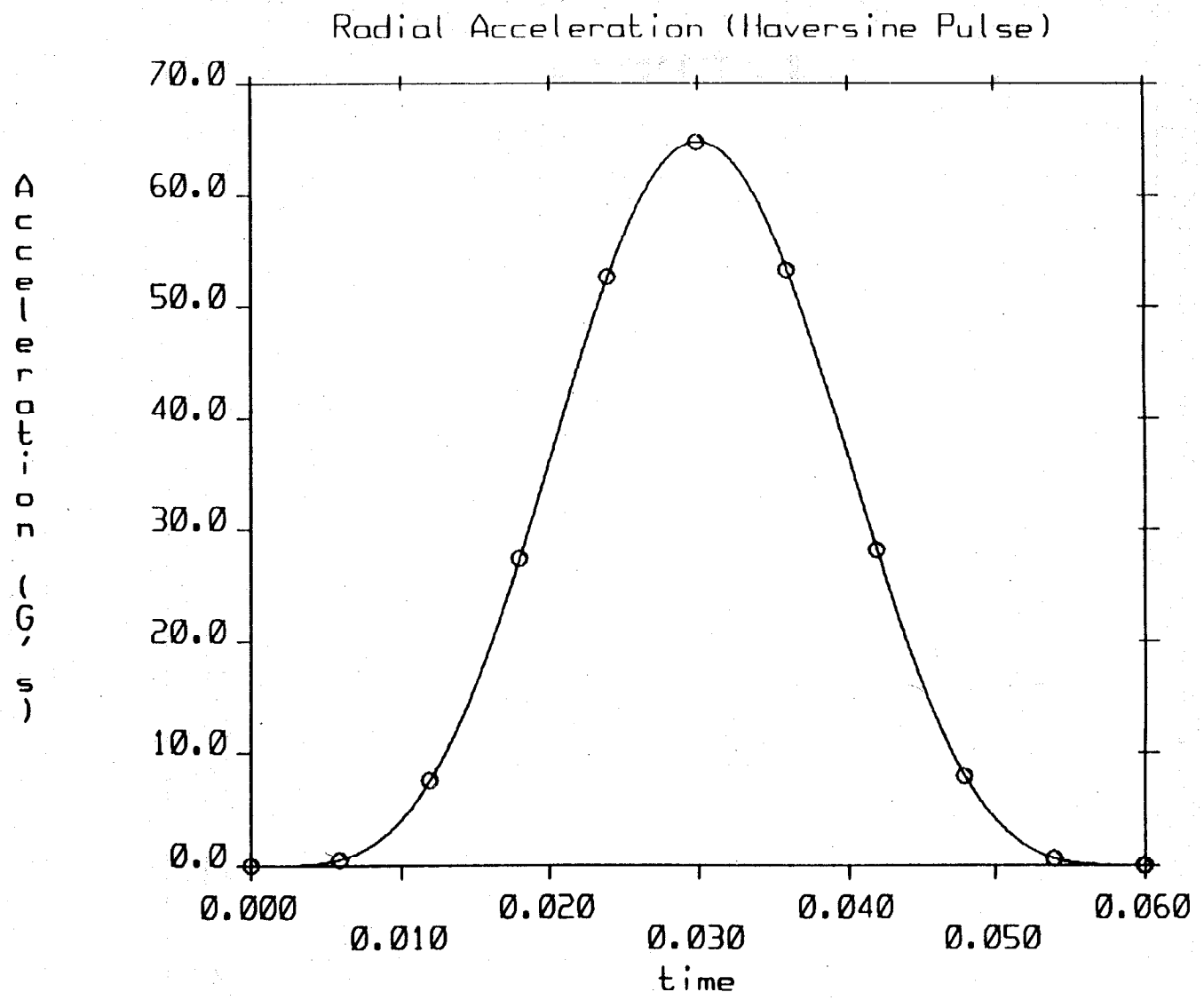


FIGURE 4-14. ANGULAR ACCELERATION TIME HISTORY HAVERSINE PULSE

4.5.3 Comparative Test Scenario

To check the transformation process, ideal or coplanar geometry (refer to Table 4-1) was used in both generating and transforming the accelerometer data, since this removes potential errors associated with small differences in accelerometer position within the array. To assess the effect of non-coplanar geometry in terms of final position of the rotated object, accelerometer data was then generated using generalized non-coplanar geometry (refer to Table 4-2) and transformed using the ideal or coplanar geometry assumption. To assess effects on strains induced in soft deformable material, failed volume calculations were also made for this condition. Finally, to determine how much of this difference in final position could be removed, accelerometer data was generated using generalized non-coplanar geometry and transformed using the coplanar assumption and an algorithm which compensates for variations from ideal geometry.

4.5.4 Results — Ideal Geometry

Figure 4-12 illustrates the location of each accelerometer within the NAP array after sequential rotations about the body x, y, and z axes of 0.5 radians, assuming either ideal or general non-coplanar geometry for the non-coplanar 3-2-2-2 configuration. Simulated acceleration signals at each location within the NAP array were computed using ideal geometry, and are shown in Figures 4-15 through 4-17 based on the haversine angular velocity pulse described above. These acceleration time histories were then transformed using the NAP software (which was modified to reflect the new initial relationship between the inertial and body coordinate systems after body rotations of 0.5 radians about each body axis), which also produced load curves for driving the anatomic model.

The model was run with a second headpart positioned at the final angular position for comparison with the headpart which was driven by load curve data based on the simulated accelerometer data. Angular velocities about the body x, y, and z axes are shown in Figure 4-18 as computed with the NAP software package. Corresponding angular velocities transformed to the fixed coordinate system are shown in Figure 4-19, and indicate that the angular velocities about the x and z body axes are reduced to zero in the fixed coordinate system, as required for the input condition of rigid body rotation solely about the fixed Y or lateral, axis. Accelerations at the CG are shown in Figures 4-20 and 4-21 in the body and fixed coordinate systems respectively, and properly indicate that the Y component of acceleration is reduced to zero in the fixed coordinate system.

Figure 4-22 illustrates headpart kinematic response, in 20 msec intervals, from time zero to 60 msec. In this figure, the headpart driven by the load curves is rotated about a fixed axis, 10 inches below the headpart CG, into a second headpart which is rotated 85.94 degrees about the center of rotation. Figure 4-23 provides several other views of the two overlaid headparts at 60 msec. Figures 4-22 and 4-23 indicate that the transformation to

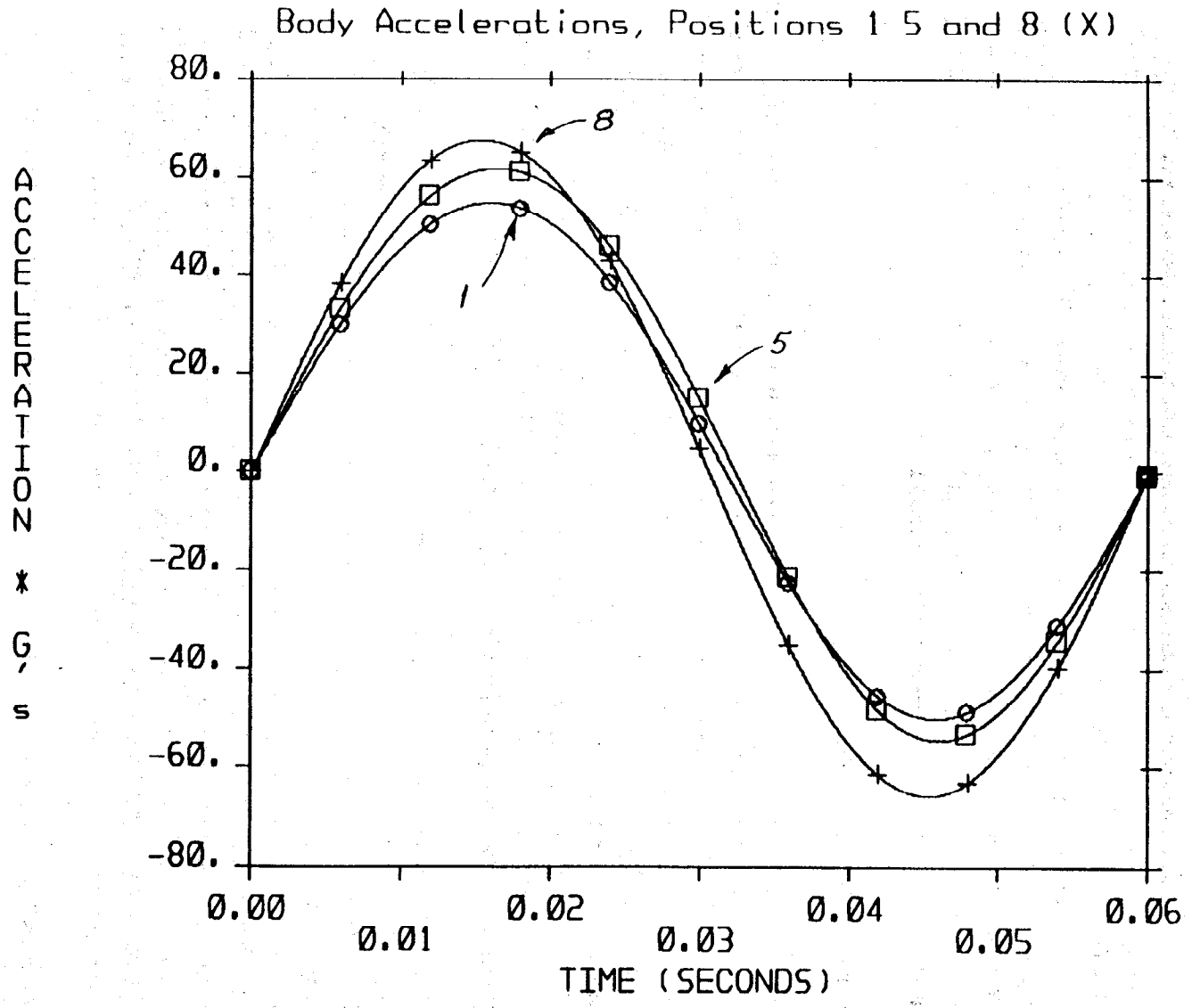


FIGURE 4-15. SIMULATED NAP X AXIS ACCELERATIONS (NOS 1, 5, AND 8) (IDEAL GEOMETRY - ROTATED AXES)

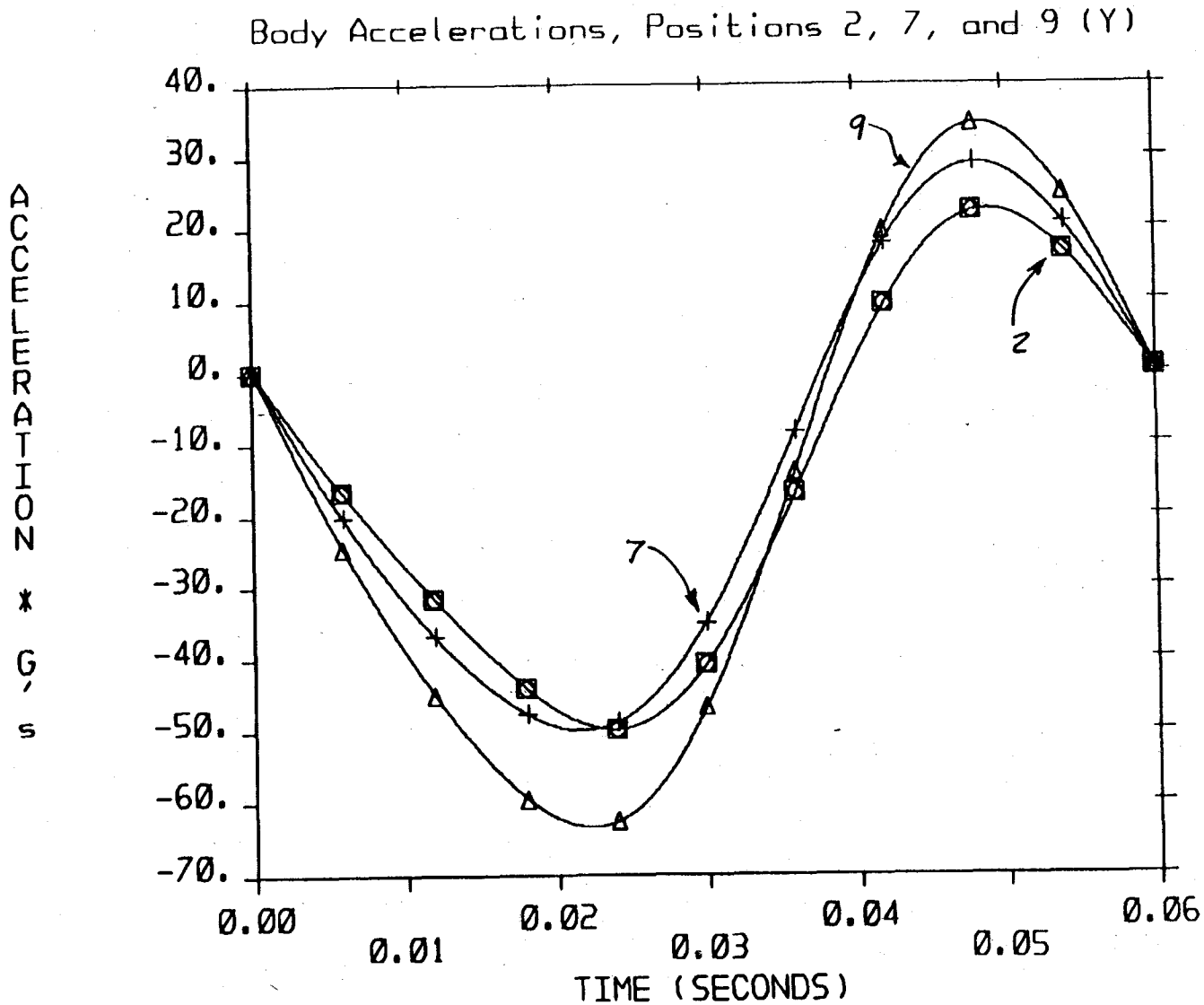


FIGURE 4-16. SIMULATED NAP Y AXIS ACCELERATIONS (NOS 2, 7, AND 9) (IDEAL GEOMETRY - ROTATED AXES)

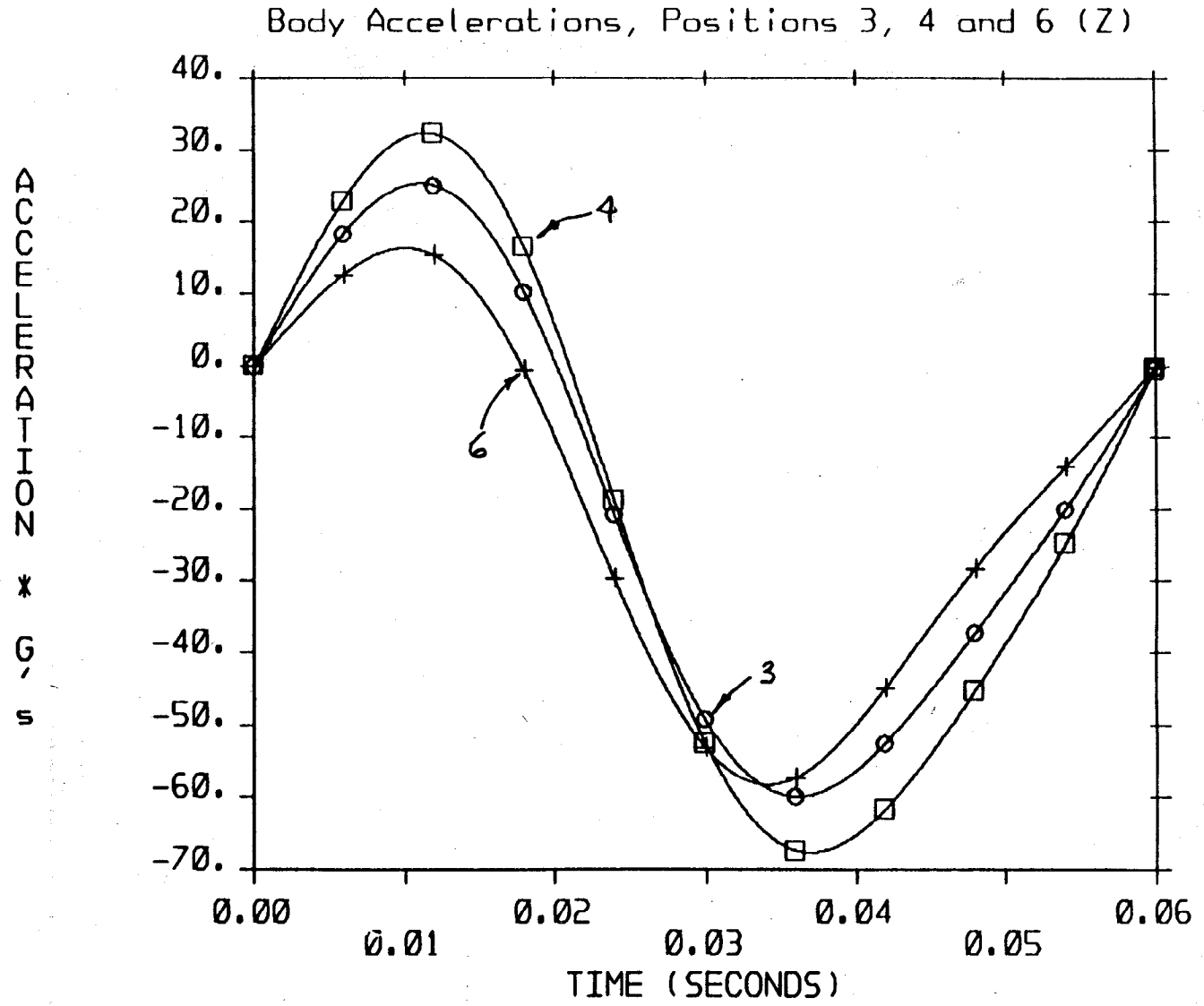


FIGURE 4-17. SIMULATED NAP Z AXIS ACCELERATIONS (NOS 3, 4, AND 6) (IDEAL GEOMETRY - ROTATED AXES)

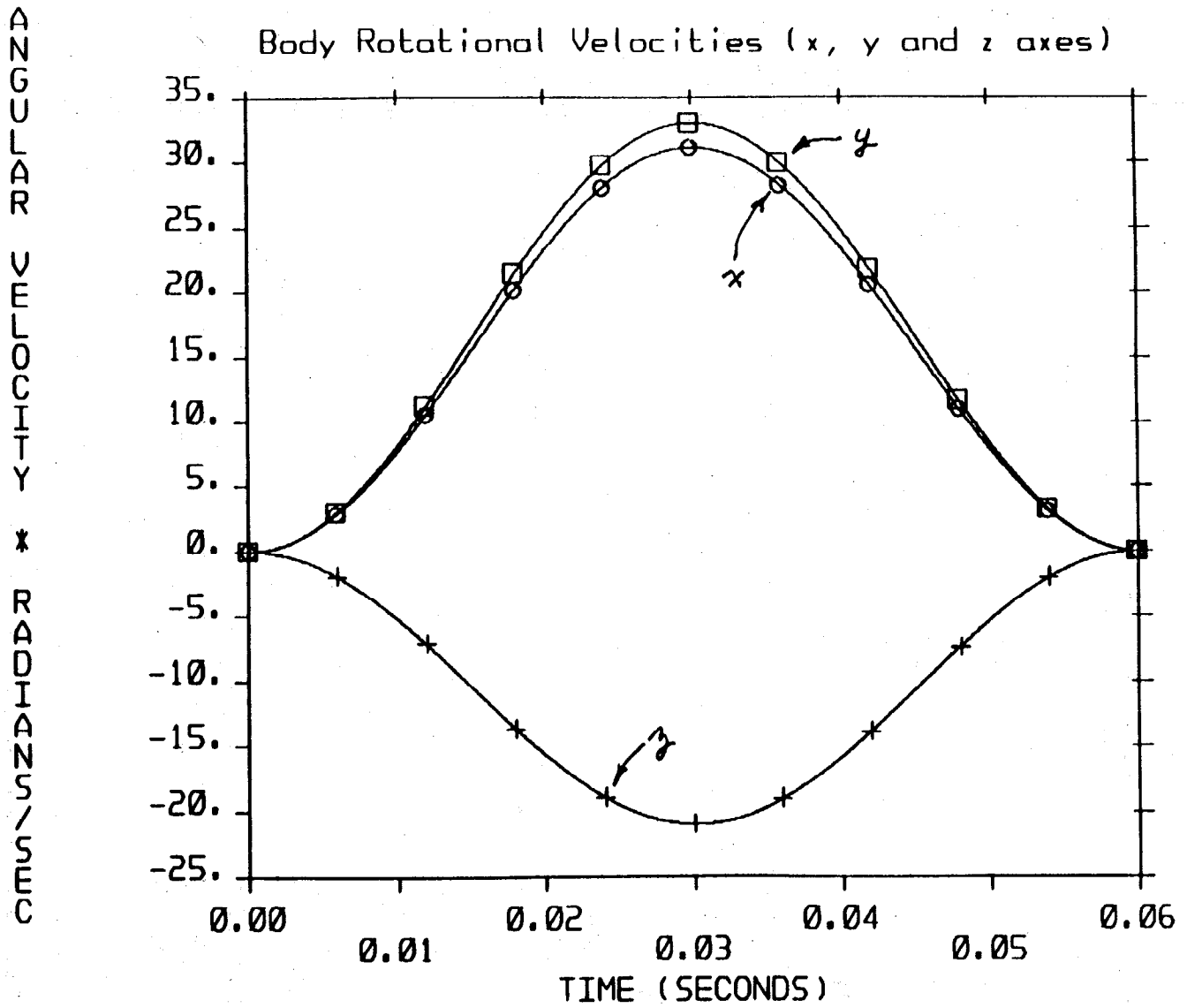


FIGURE 4-18. ANGULAR VELOCITIES ABOUT BODY X, Y, AND Z AXES (IDEAL GEOMETRY - ROTATED AXES)

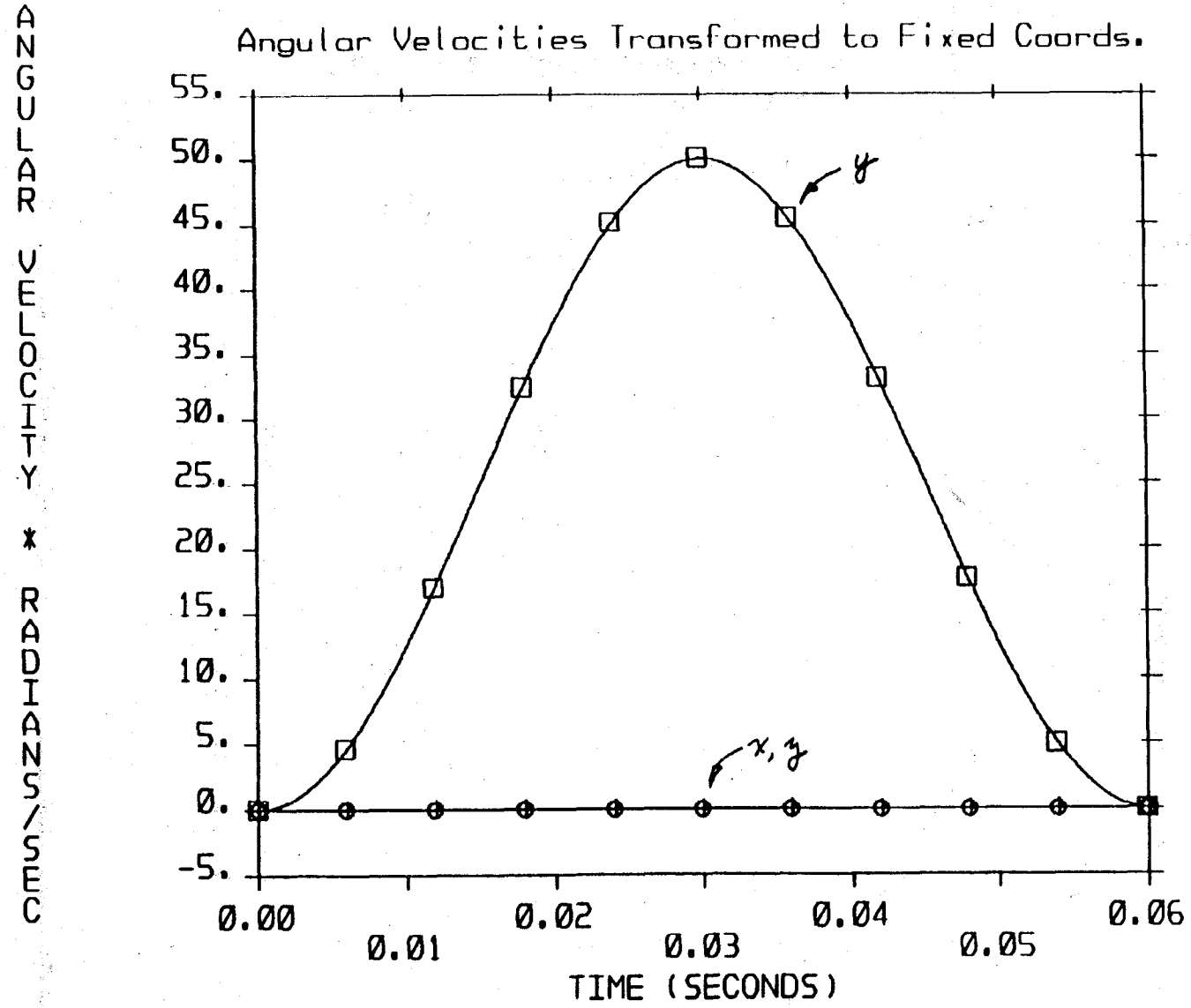
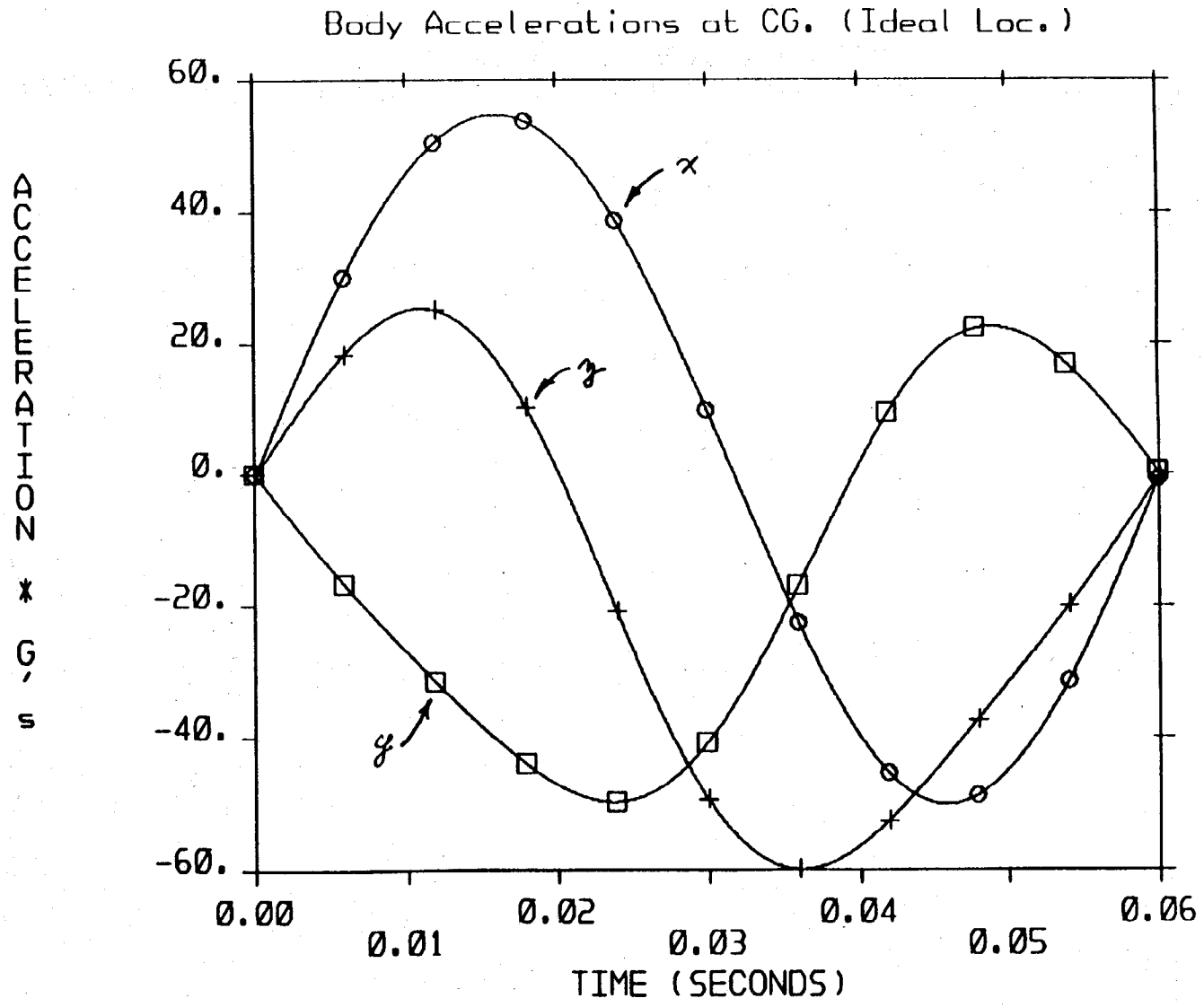


FIGURE 4-19. ANGULAR VELOCITIES TRANSFORMED TO FIXED COORDINATE SYSTEM (IDEAL GEOMETRY - ROTATED AXES)



FIGURES 4-20. X, Y, Z BODY ACCELERATIONS AT CG LOCATION (IDEAL GEOMETRY - ROTATED AXES)

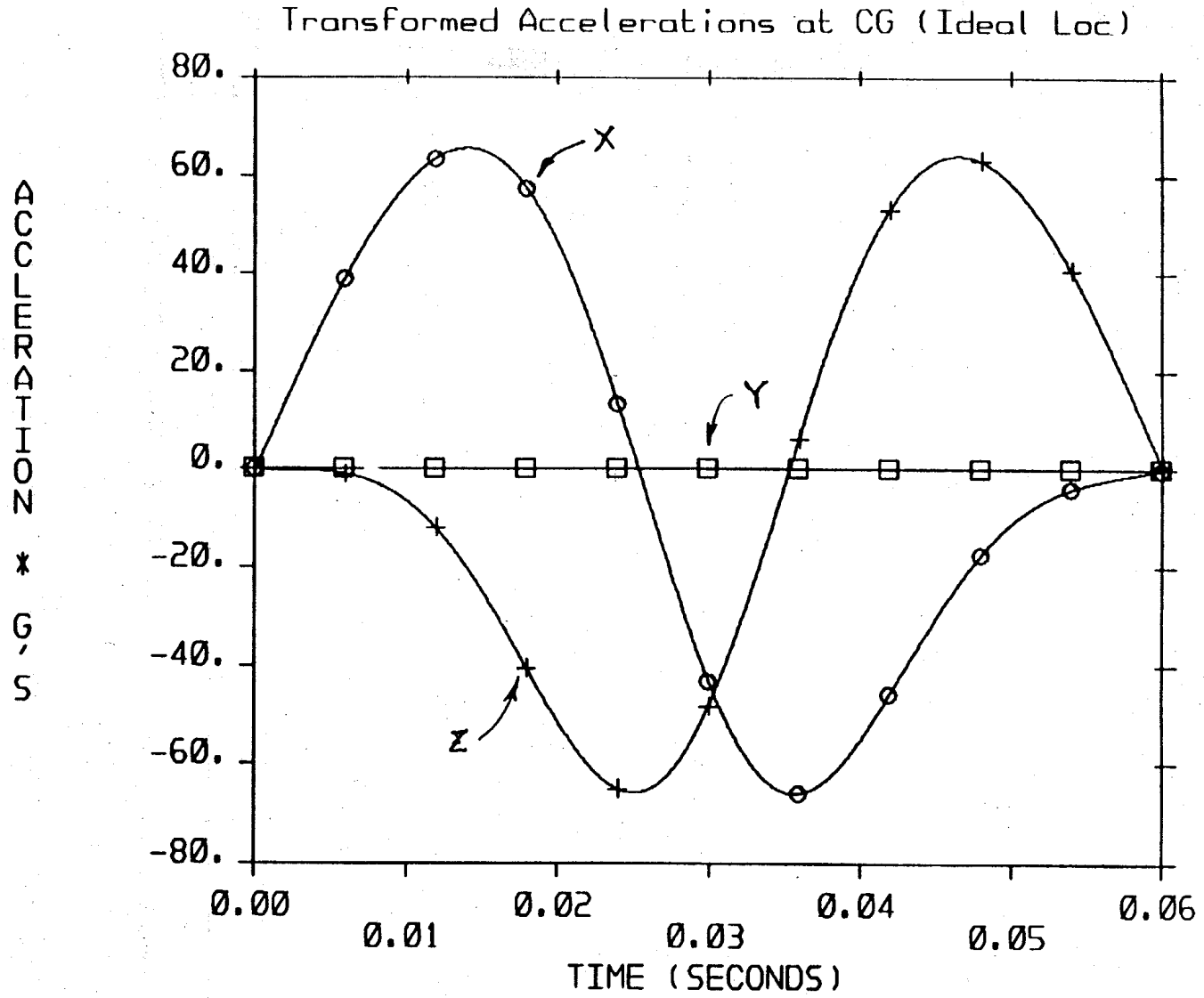


FIGURE 4-21. CG ACCELERATIONS TRANSFORMED TO FIXED COORDINATE SYSTEM (IDEAL GEOMETRY - ROTATED AXES)

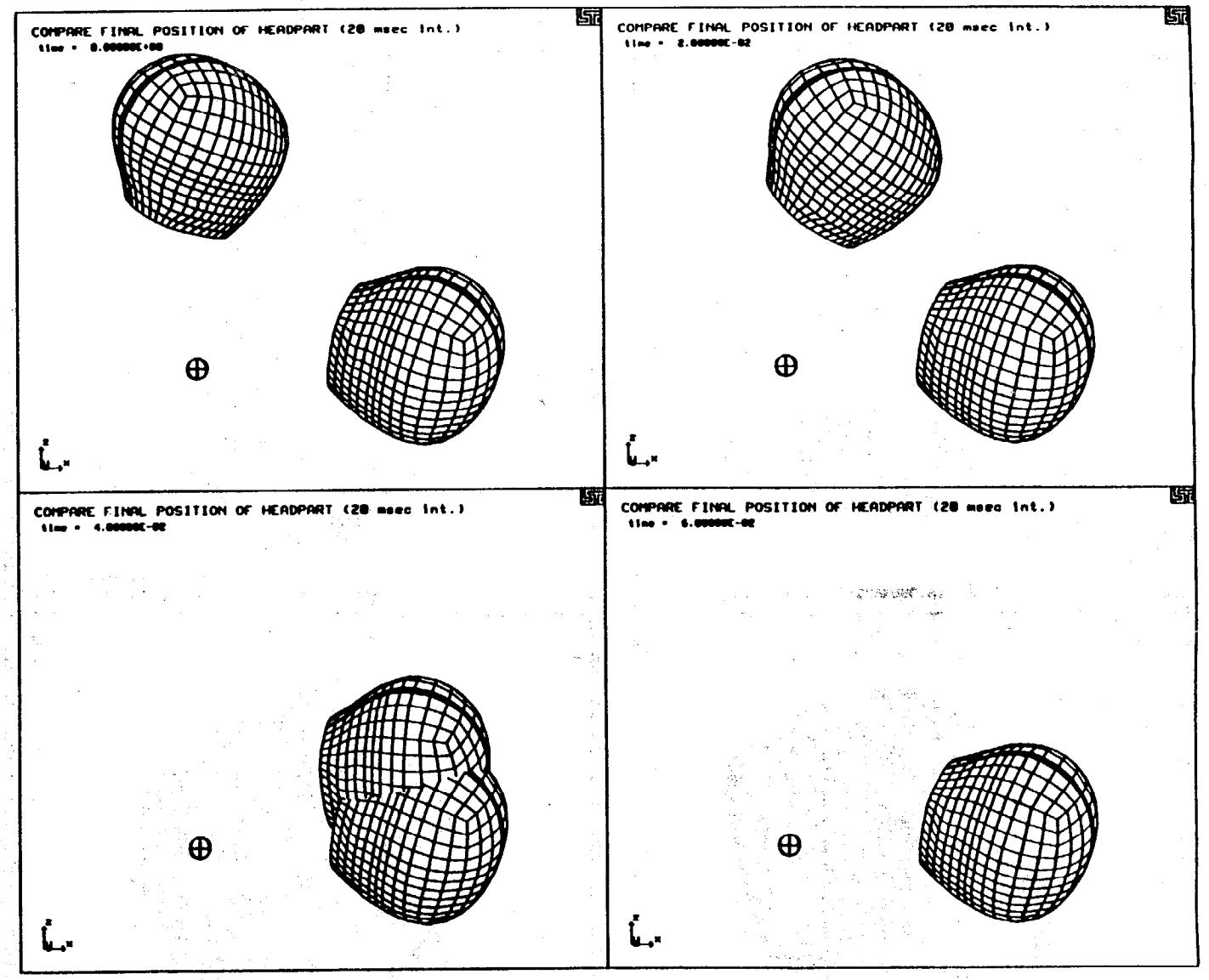


FIGURE 4-22. HEADPART KINEMATIC RESPONSE IN 20 MSEC INTERVALS (IDEAL GEOMETRY - ROTATED AXES)

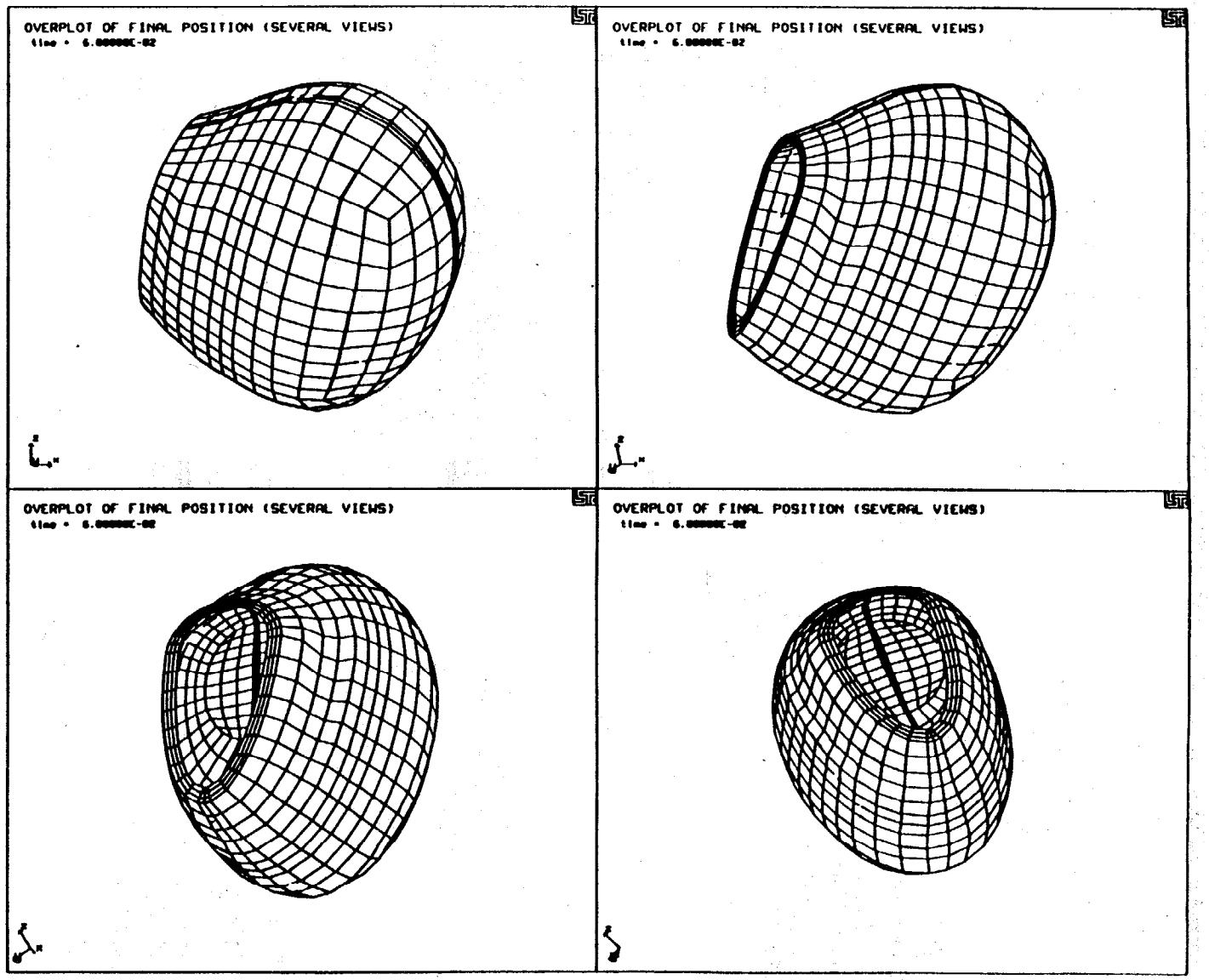


FIGURE 4-23. COMPARISON OF HEADPART LOCATION WITH CORRECT FINAL POSITION (IDEAL GEOMETRY - ROTATED AXES)

the final position is accomplished virtually without error in final position. Failed volume data, indicating the fraction of soft, deformable interior material exceeding specific strain levels, is shown in Figure 4-24 for strain values of 4, 6, 8, 10, and 12 percent strain. These results are compared and discussed below, with similar results using a generalized non-coplanar NAP geometry.

4.5.5 Results — General Non-Coplanar Geometry

Figure 4-25 illustrates the location of each accelerometer within the NAP array after sequential rotations about the body x, y and z axes of 0.5 radians, for the general non-coplanar 3-2-2-2 configuration (refer to Table 4-2). For this geometry, simulated acceleration signals at each rotated location along the body x, y and z axes are shown in Figures 4-26 through 4-28 for the haversine angular velocity pulse described above. These acceleration time histories were then transformed using the NAP software which assumes ideal location of accelerometers within the NAP array (refer to Table 4-1). This simulation was made to characterize the error in final position, differences in kinematic response variables, and the effect on computed strain, produced for the condition of taking test data from a general non-coplanar NAP geometry, and transforming these signals assuming ideal geometry conditions.

Again, load curves generated for the above case were used to drive the anatomic model, and a second headpart was located at the final angular position for comparison. From the transformation process, angular velocities about the body x, y and z axes are shown in Figure 4-29. Corresponding angular velocities transformed to the fixed coordinate system are shown in Figure 4-30, and are not reduced to zero as in the ideal geometry case. The peak value of transformed angular velocity about the fixed X and Z axes is about -2.5 rad/sec, or about 5 percent of the correct maximum angular velocity of 50 rad/sec about the fixed Y axis, while the transformed angular velocity about the fixed Y axis is about 52 rad/sec, or about 4 percent high.

For the general non-coplanar geometry condition, a comparison of headpart kinematic response in 20 msec intervals is illustrated in 4-31, overlaid with the second headpart located at the known final location. Several other views of the two overlaid headparts at 60 msec are contained in Figure 4-32. Figures 4-33 and 4-34 illustrate body accelerations for the three accelerometers located at the CG and Figures 4-35 and 4-36 illustrate displacement of the CG along the fixed X, Y, and Z axes, for the ideal versus actual geometry conditions, respectively. These figures may be compared to illustrate variations in translational kinematics resulting from the non-coplanar location of accelerometers in the NAP array.

Failed volume data, indicating the volume fraction of deformable material exceeding specific strain levels, is shown in Figure 4-37 for strain values of 4, 6, 8, 10, and 12

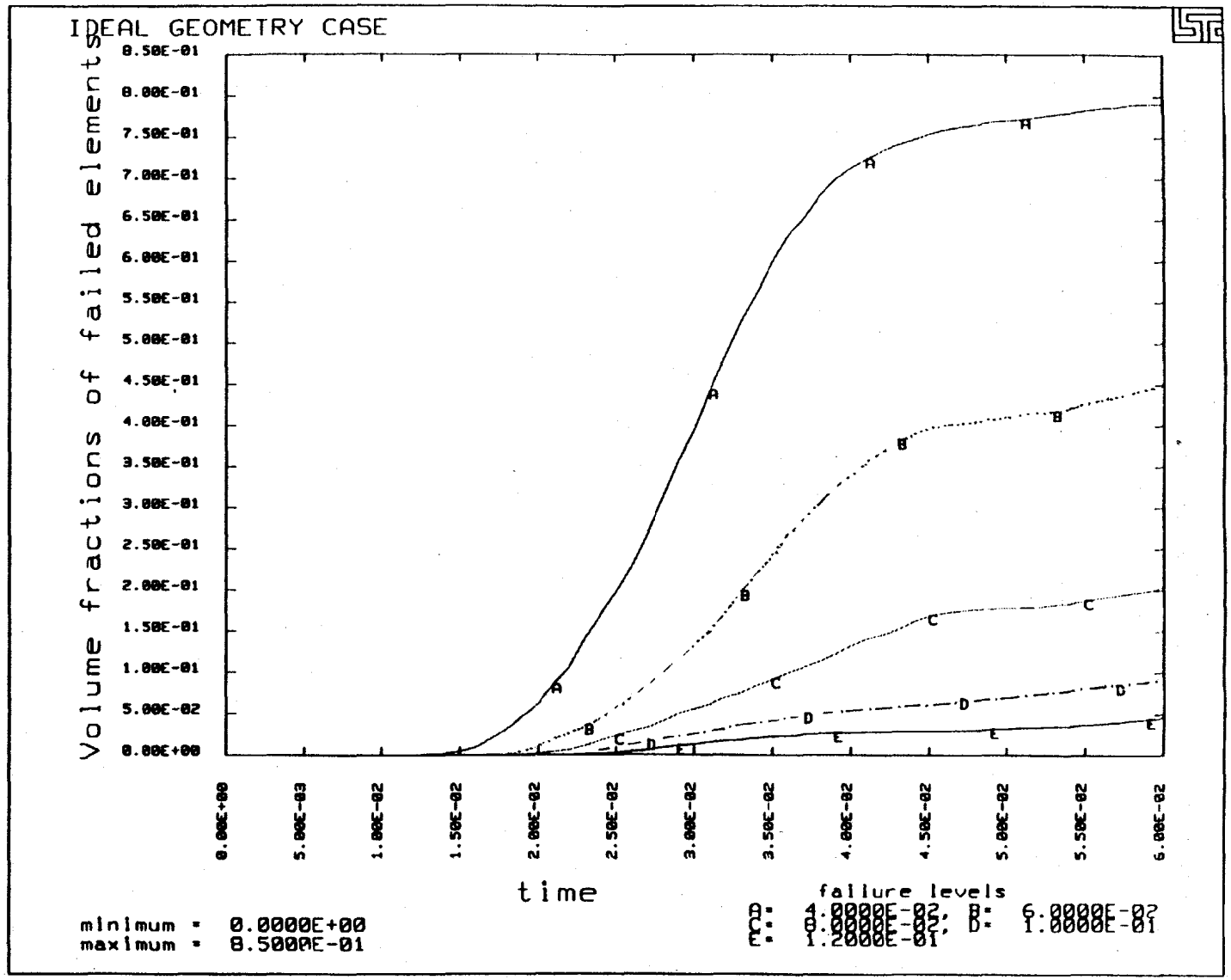


FIGURE 4-24. VOLUME FRACTION OF FAILED ELEMENTS VS STRAIN LEVEL (IDEAL GEOMETRY - ROTATED AXES)

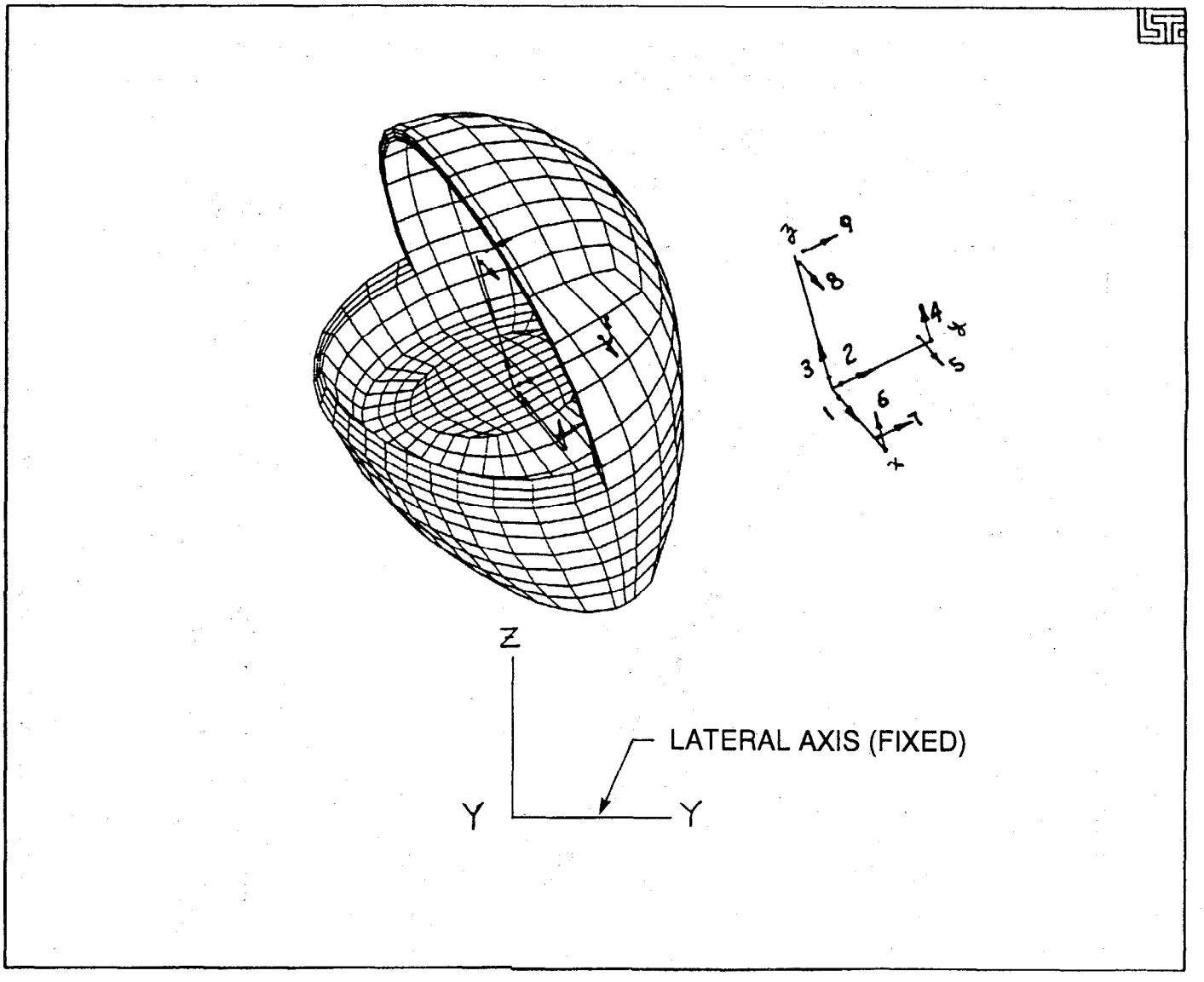


FIGURE 4-25. ACCELEROMETER LOCATIONS IN ROTATED POSITION BASED ON NON-COPLANAR 3-2-2-2 GEOMETRY

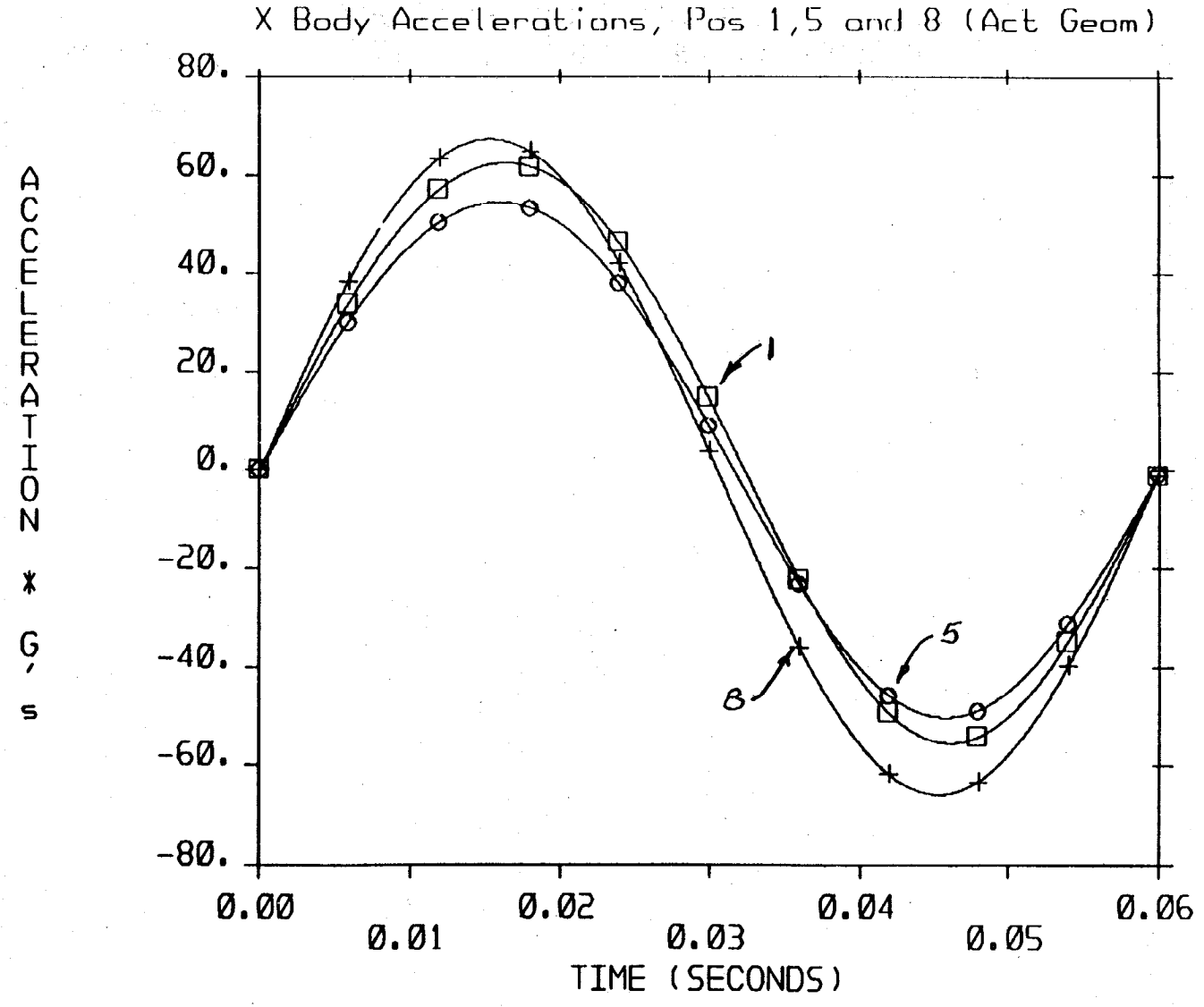


FIGURE 4-26. SIMULATED NAP X AXIS ACCELERATIONS (NOS 1, 5, AND 8) (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

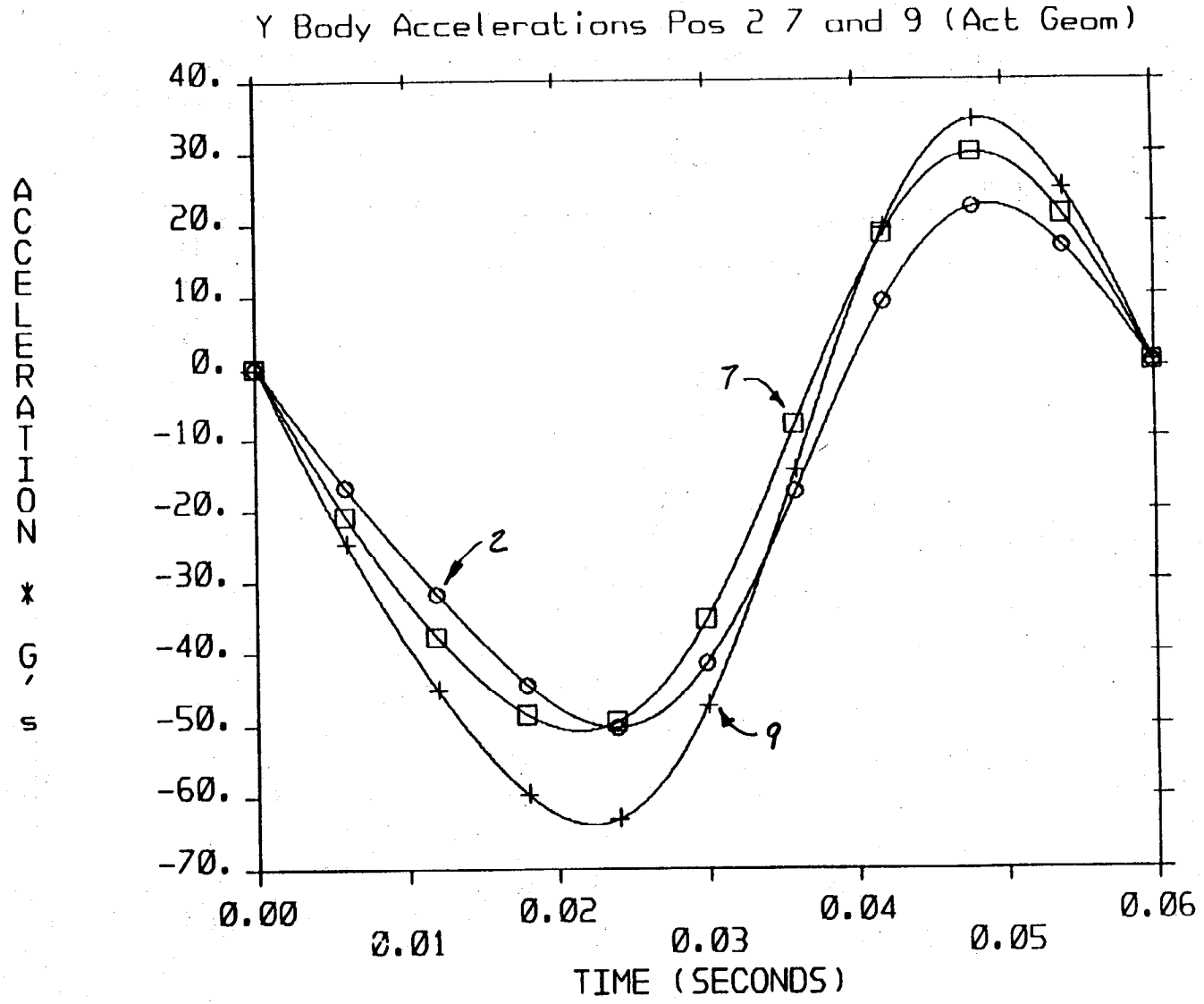


FIGURE 4-27. SIMULATED NAP Y AXIS ACCELERATIONS (NOS 2, 7, AND 9) (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

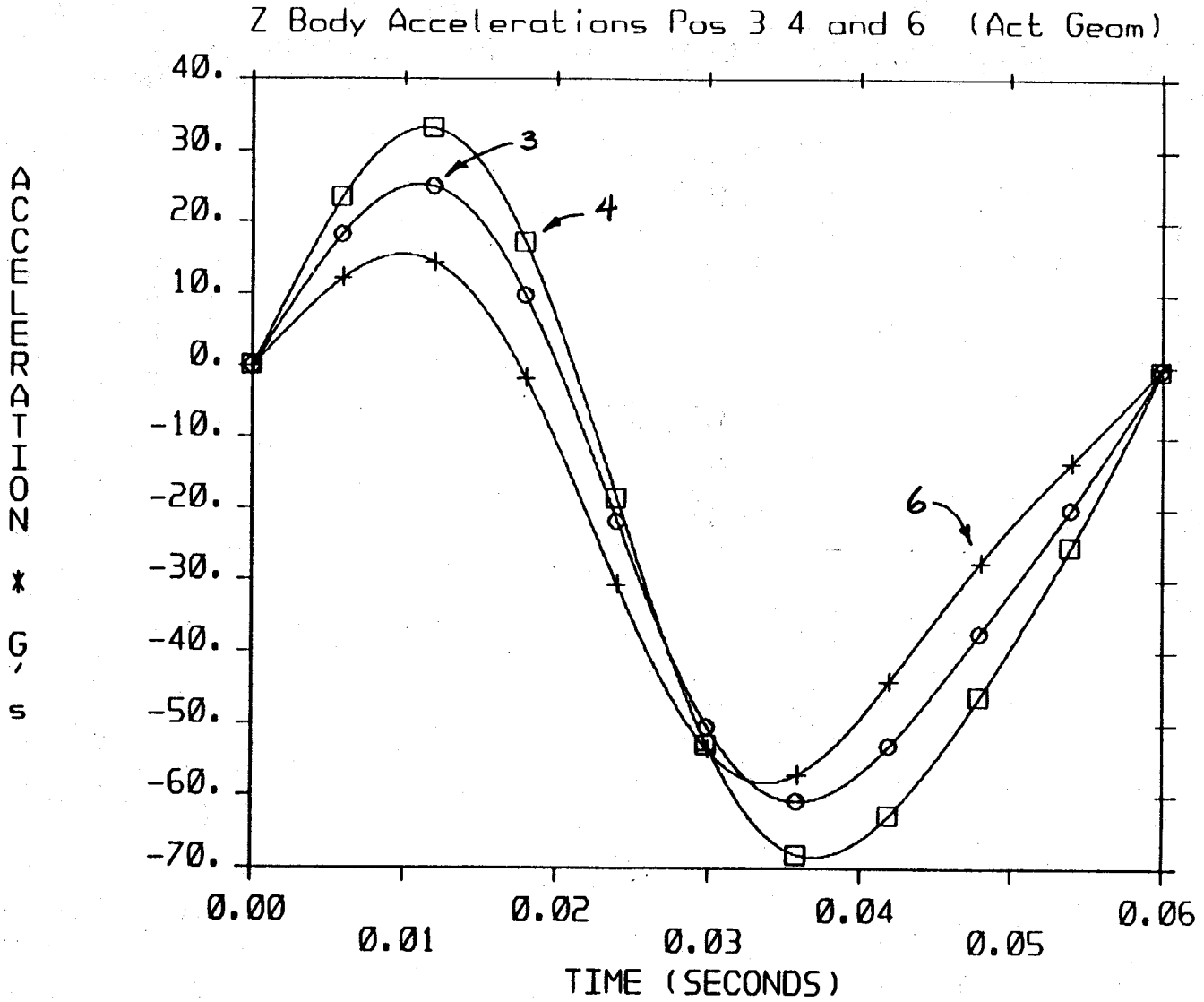


FIGURE 4-28. SIMULATED NAP Z AXIS ACCELERATIONS (NOS 3, 4, AND 6) (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

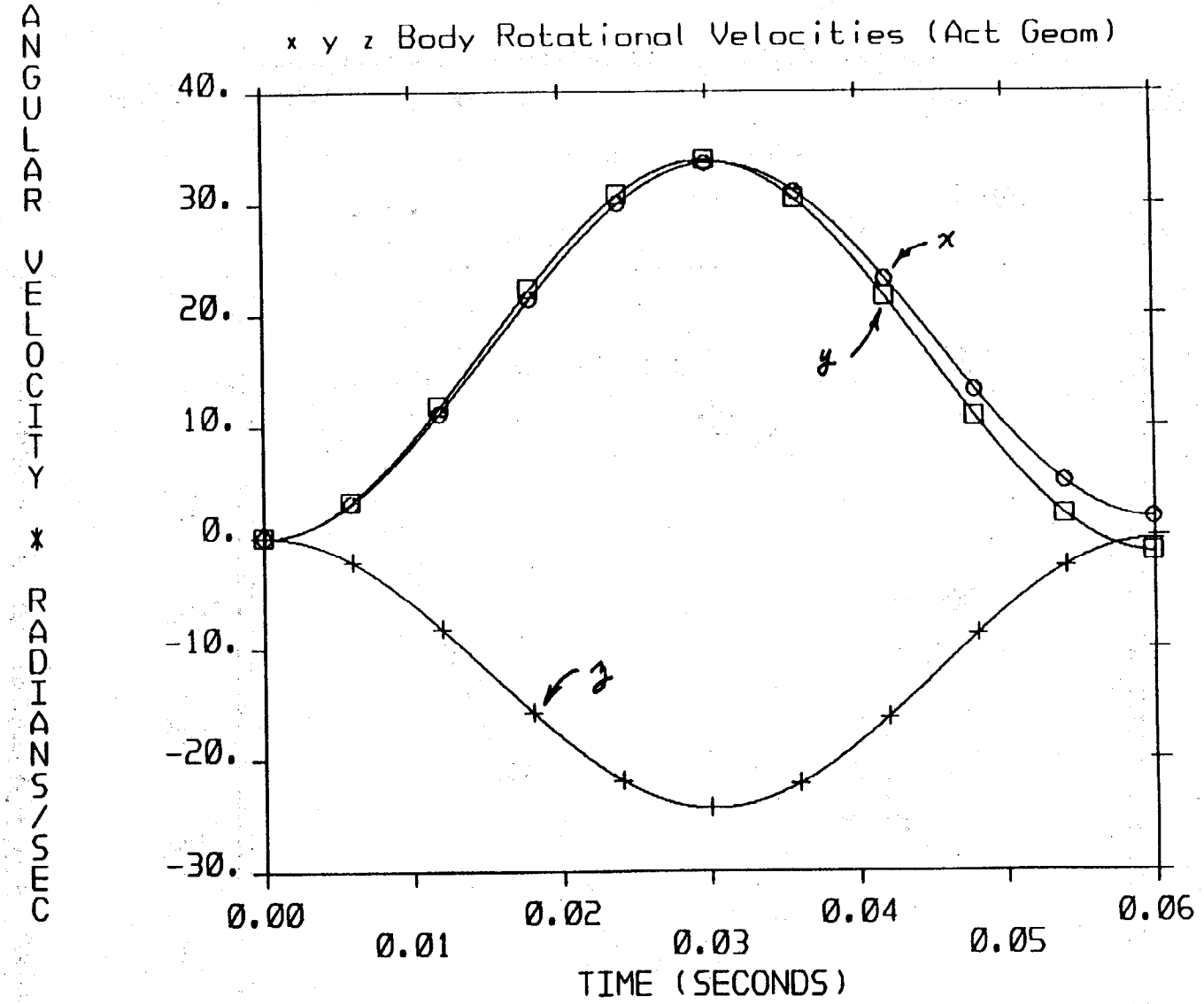


FIGURE 4-29. ANGULAR VELOCITIES ABOUT THE BODY X, Y, AND Z AXES (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

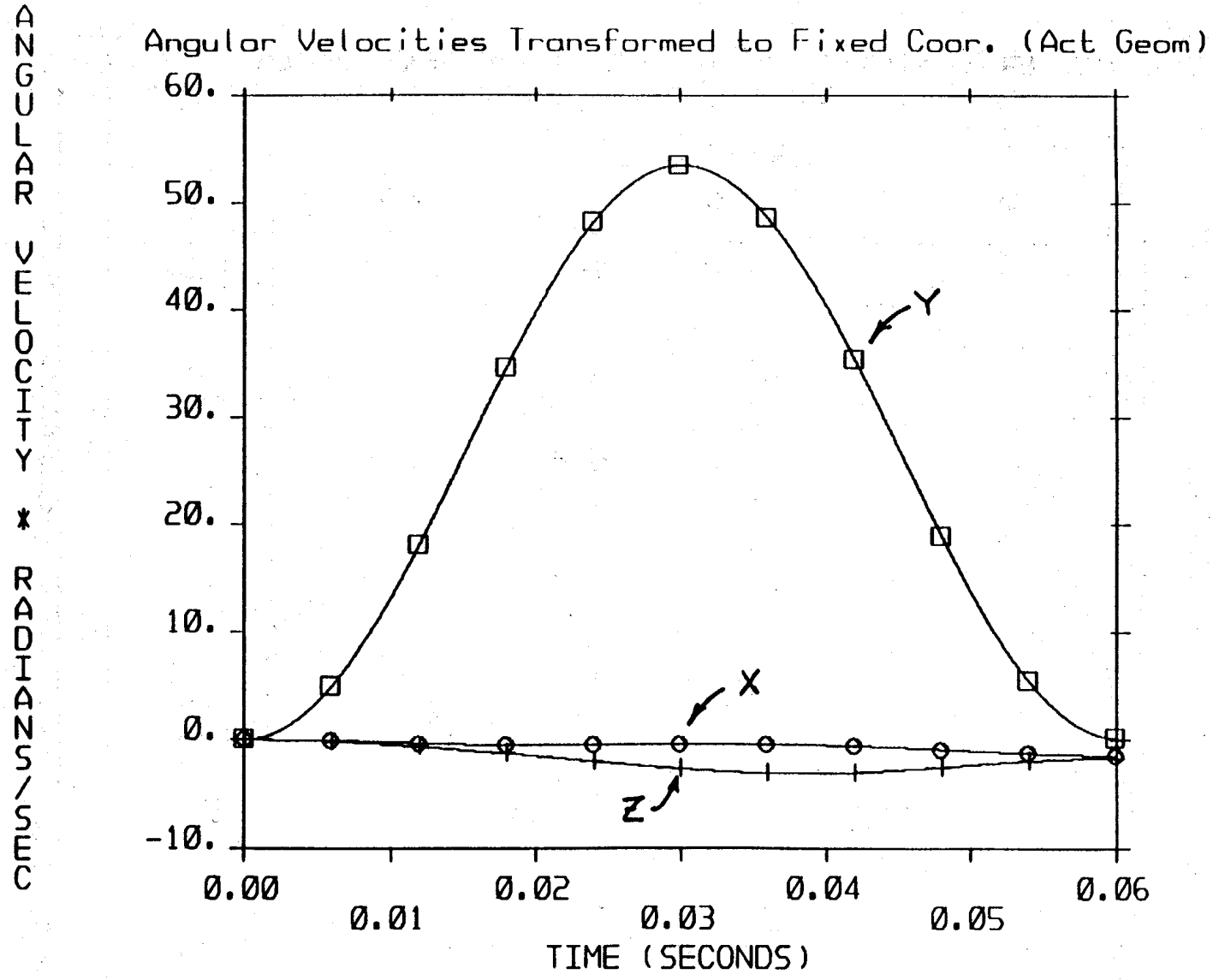


FIGURE 4-30. ANGULAR VELOCITIES TRANSFORMED TO FIXED COORDINATES (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

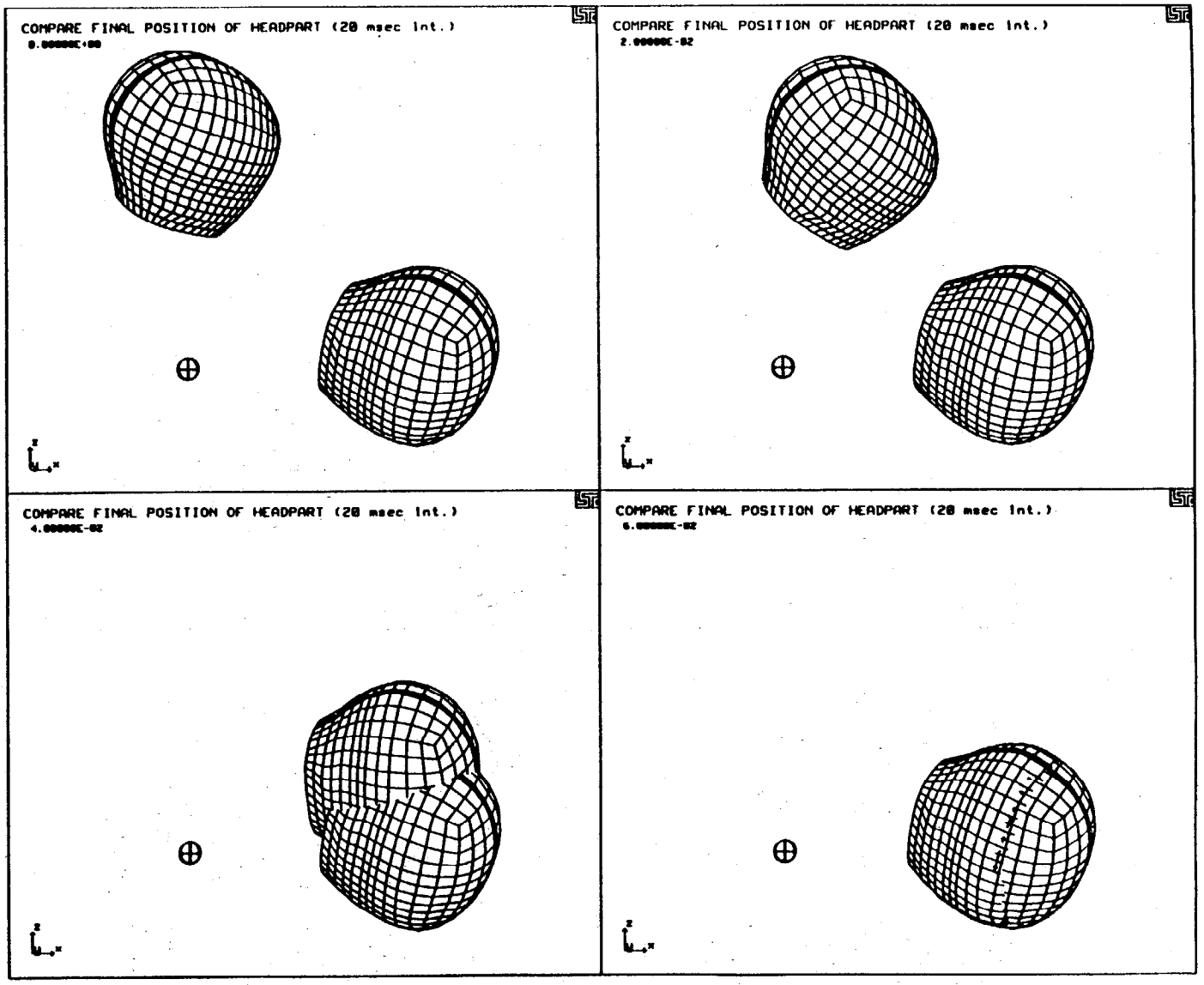


FIGURE 4-31. HEADPART KINEMATIC RESPONSE IN 20 MSEC INTERVALS (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

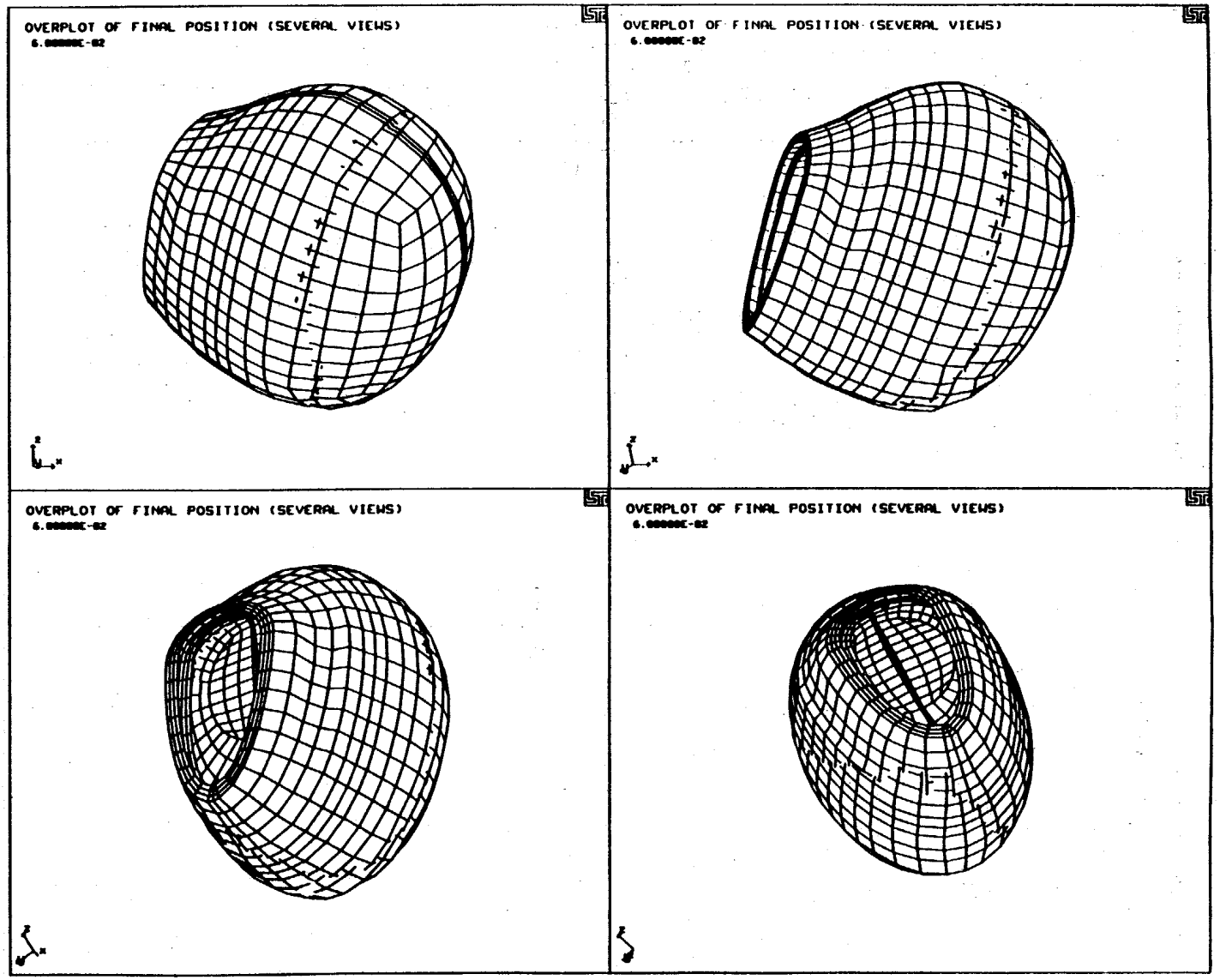


FIGURE 4-32. COMPARISON OF HEADPART LOCATION WITH CORRECT FINAL POSITION (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

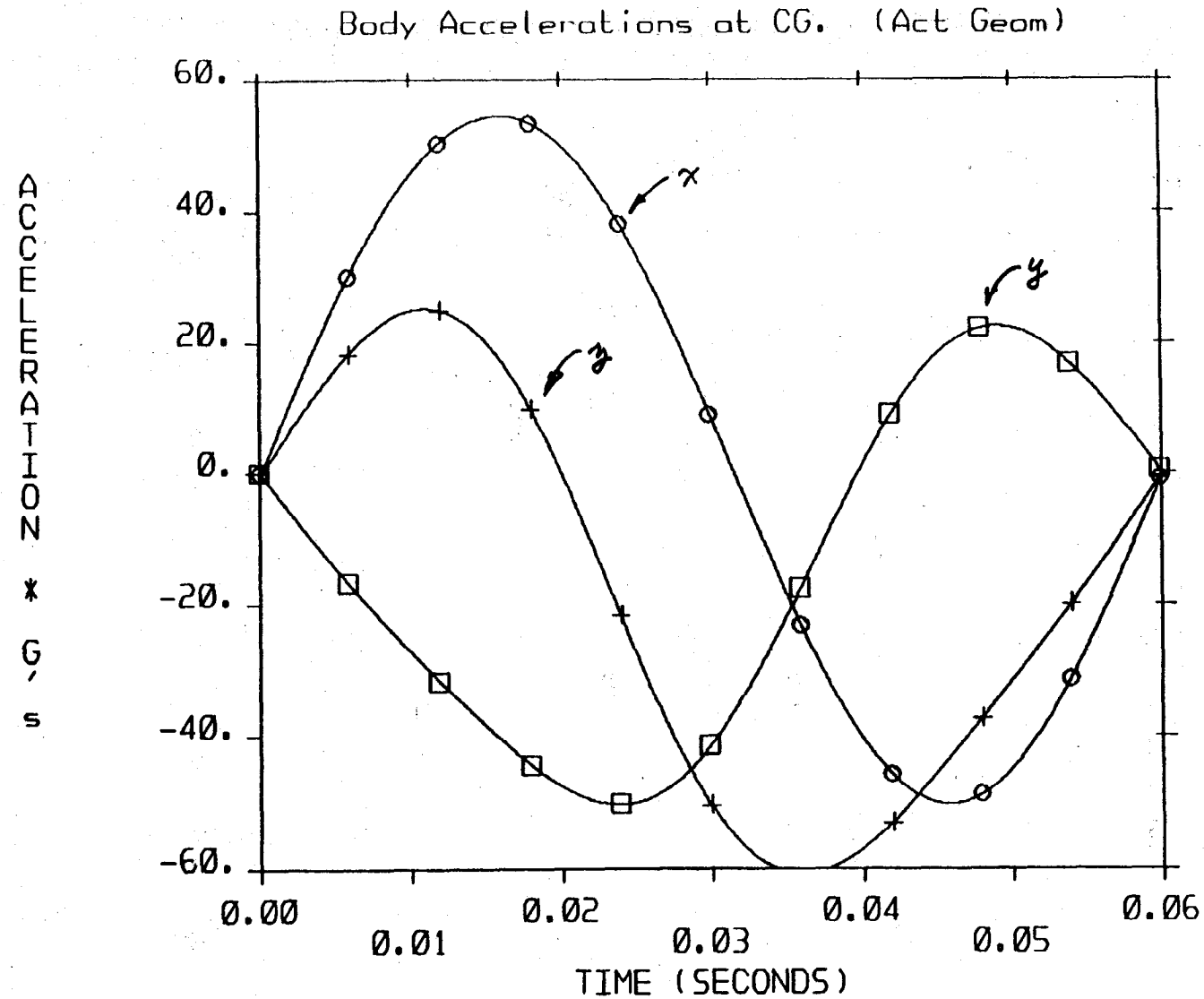


FIGURE 4-33. X, Y, Z BODY ACCELERATIONS AT CG (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

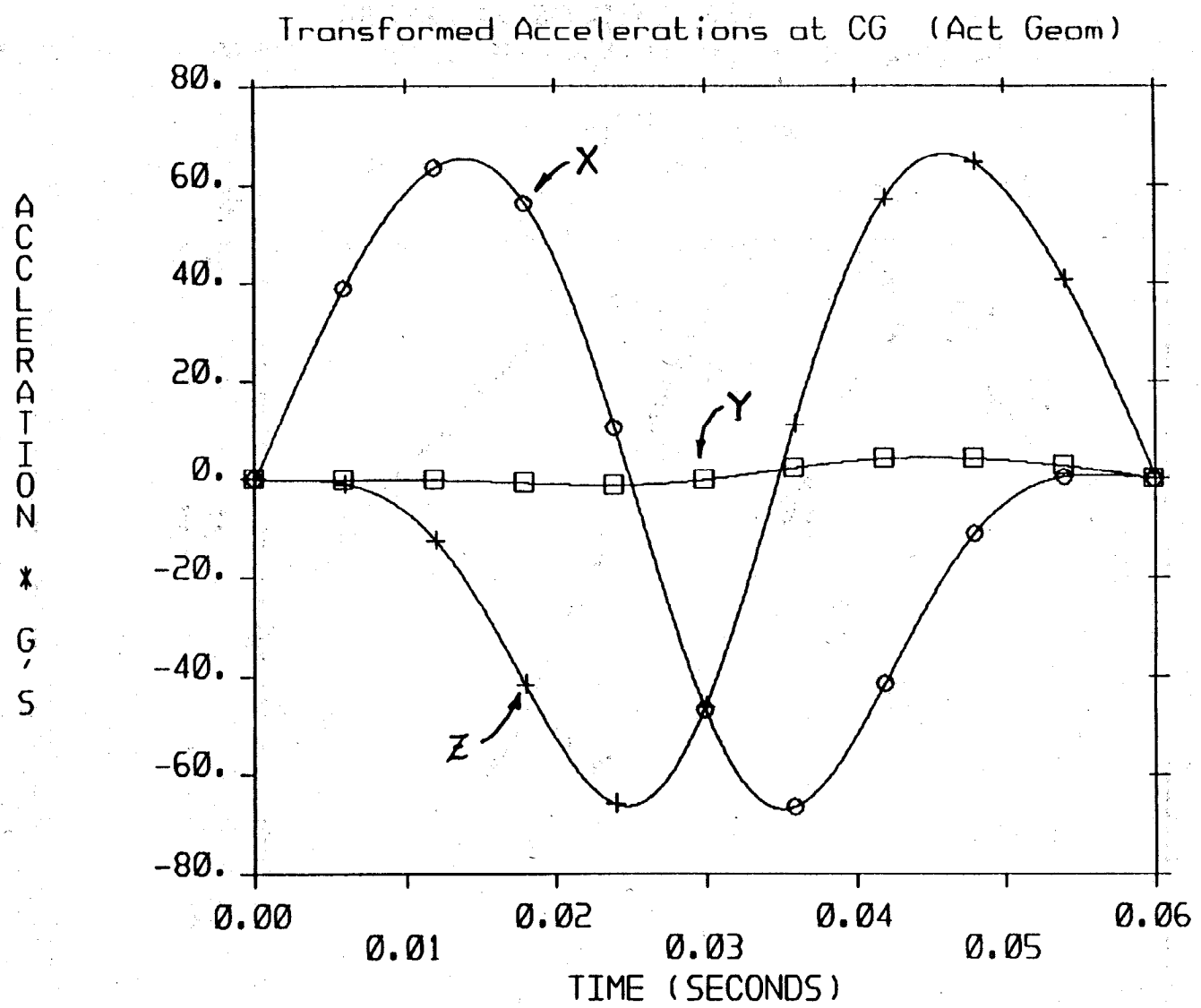


FIGURE 4-34. X, Y, Z ACCELERATIONS TRANSFORMED TO FIXED COORDINATES (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

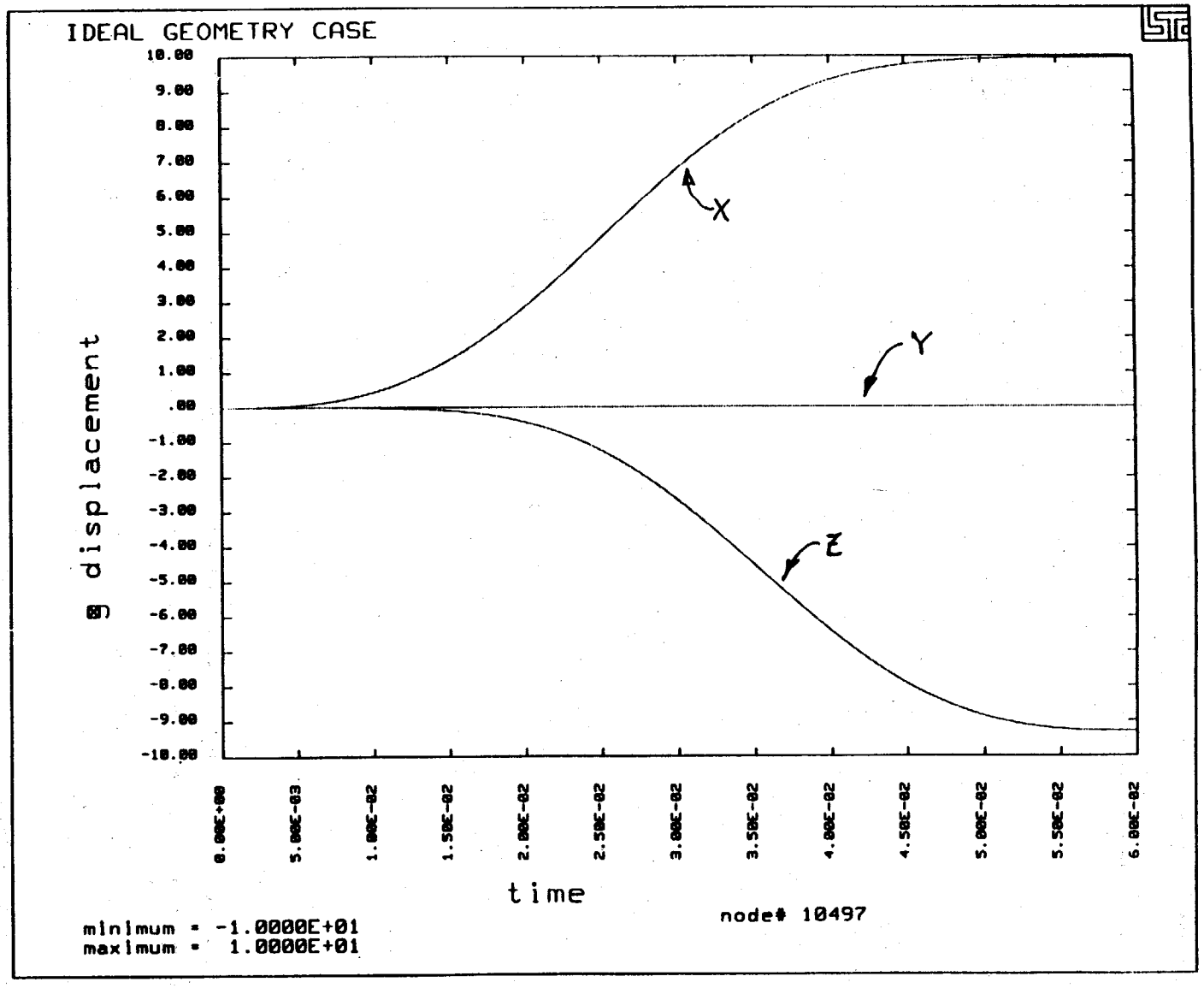


FIGURE 4-35. TRANSLATIONAL DISPLACEMENTS ALONG FIXED X, Y, AND Z AXES (IDEAL GEOMETRY CONDITION)

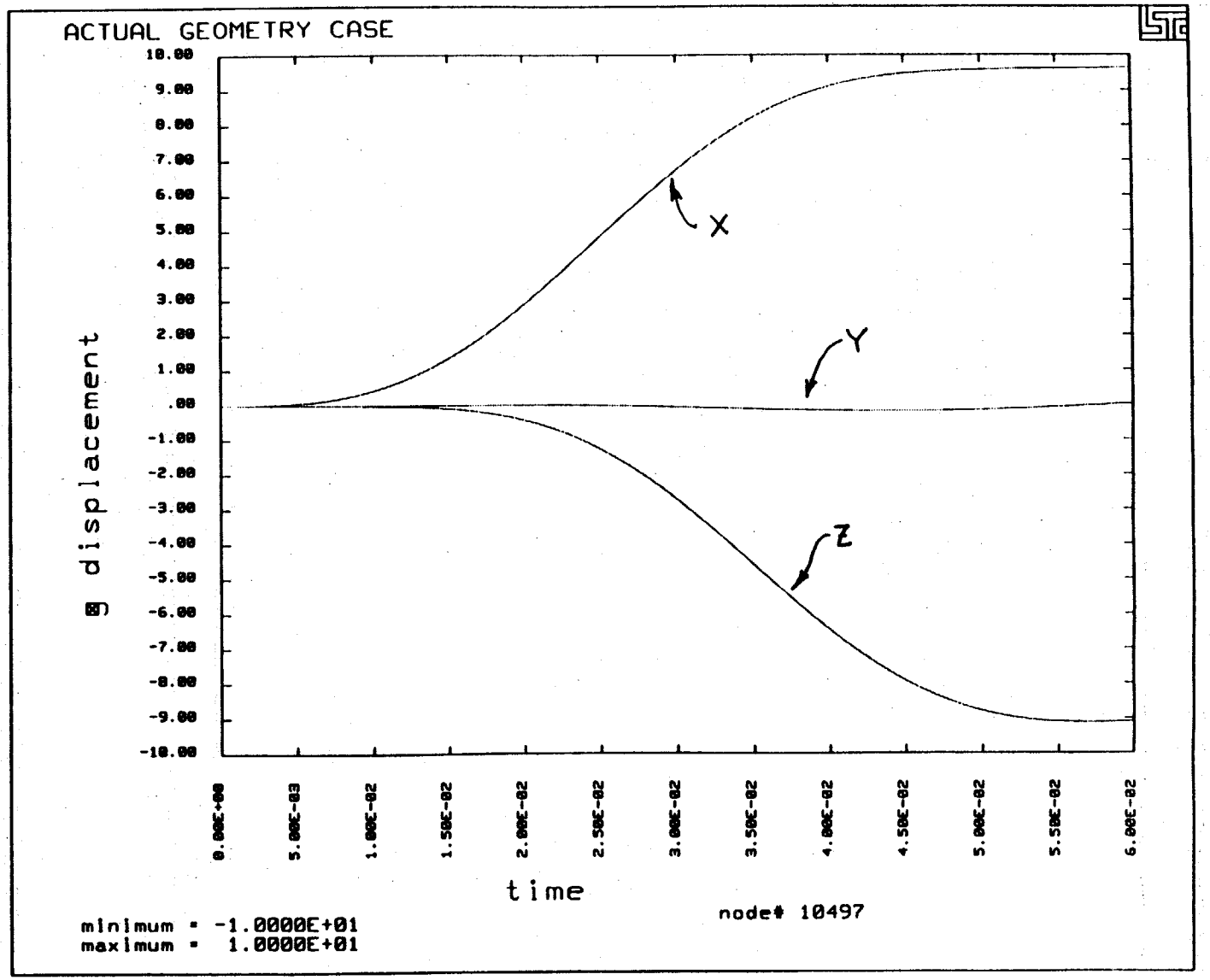


FIGURE 4-36. TRANSLATIONAL DISPLACEMENT ALONG FIXED X, Y, AND Z AXES (GENERAL NON-COPLANAR GEOMETRY CONDITION)

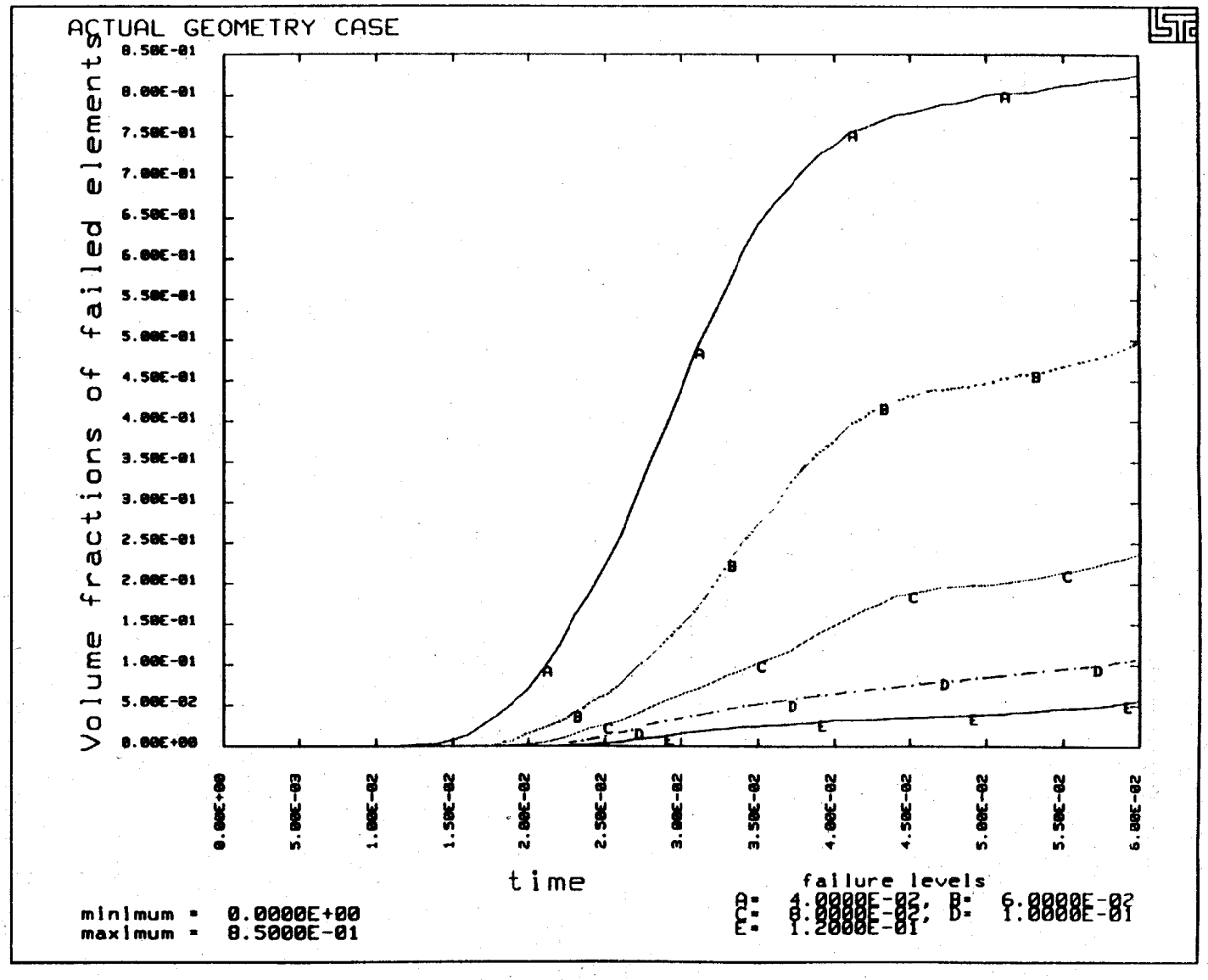


FIGURE 4-37. VOLUME FRACTION OF FAILED ELEMENTS (GENERAL NON-COPLANAR GEOMETRY - ROTATED AXES)

percent strain. A comparison of failed volume strain levels for the general non-coplanar geometry vs the ideal geometry condition discussed above, follows in Table 4-3.

TABLE 4.3. COMPARISON OF VOLUME FRACTION OF FAILED ELEMENTS FOR SELECTED STRAIN LEVELS IDEAL VS GENERAL NON-COPLANAR GEOMETRY CONDITION

Strain level	0.04	0.06	0.08	0.10	0.12
Ideal geom	79%	45%	20.1%	9.0%	4.8%
Non-Coplanar	82.5%	49%	23.6%	10%	5.5%
Difference	3.5%	4%	3.5%	1%	0.7%
% Variation	4.4	8.9	17.4	11.1	14.6

For each strain level, these results indicate (a) differences in the computed volume fraction of failed elements and (b) percent variation of failed volumes for the ideal (i.e. exact) vs general non-coplanar geometry condition. Differences in the computed volume fractions of failed elements range from 4% for lower strain levels, to 0.7% at higher strains, with percentage variations, ranging from about 4 to 17 percent. These results, along with the data presented in Figures 4-29 through 4-34, indicates that the non-coplanar location of accelerometers can have a significant influence on the resulting kinematics, and consequently failed volume strain levels, for excitations which produce relatively large angular kinematics. An algorithm for reducing these errors is discussed in the following section.

4.5.6 Compensation Algorithm for Non-Coplanar Location of Accelerometers

For perfect location of accelerometers, centripetal accelerations and cross products of angular velocity (Coriolis accelerations) relative to the origin of the NAP array are zero or can be computed without error. For non-coplanar location of accelerometers within the NAP array, correction factors for these acceleration components, including transducer measurement errors such as bias and cross-axis sensitivity, were developed as a function of offset distances from the ideal geometry configuration of the NAP array in Appendix A1, *Comparison of Translational Accelerometer Configurations for Measuring Angular Accelerations of a Rigid Body*, of Reference 4. These equations describe compensation coefficients for evaluation during a calibration procedure to provide improved estimates of angular and translational kinematics by accounting for cross products of angular velocity and centripetal accelerations produced by the non-coplanar location of accelerometers in the NAP array and can be extended to compensate for specific transducer measurement uncertainties as well.

This algorithm, developed in Appendix A, has been incorporated into the coordinate transformation software program listing of Appendix B, to improve the estimates of

angular and translational kinematics using accelerometer data from the non-coplanar NAP array. Angular accelerations are computed using a nominal (or ideal) geometry and then adjusted to include effects of centripetal accelerations and cross products of angular velocities arising due to variations from the nominal geometry, as described in Appendix A. Appendix A also describes the development of the compensation algorithm and compensation coefficients which relate estimated and actual angular kinematics to variations from coplanar geometry.

To evaluate the effectiveness of the compensation algorithm, accelerometer data was again generated for the haversine angular velocity pulse using generalized non-coplanar geometry data and transformed using the compensation algorithm. Perfect transducers are again assumed for these calculations (i.e., no errors due to measurement). Figure 4-38 illustrates angular velocity components transformed to inertial coordinates for this condition, and comparing this data with corresponding data computed for the ideal geometry condition (Figure 4-19), it can be seen that this algorithm provides virtually complete compensation for the non-coplanar geometry of the NAP array. An overlay of the final headpart position with the correct headpart position is shown in Figure 4-39 for several views. A comparison of X, Y, and Z translational displacements of the CG for this condition, shown in Figure 4-40, with the corresponding data for the ideal geometry condition shown in Figure 4-35, also indicates identical results.

Failed volume calculations show the same result. The failed volume time history data extracted for this condition is shown in Figure 4-41, and virtually overlays the calculation based on ideal geometry (Figure 4-24). Assuming perfect transducers, *these results indicate that effects of cross products of angular velocity and centripetal acceleration arising from non-ideal transducer location can be virtually eliminated during the transformation process.* A comparison of angular kinematics with the previous case indicates that errors in transformed angular velocity components are reduced by several orders of magnitude when this algorithm is applied.

These results indicate that the principal remaining uncertainty is associated with transducer-related effects such as cross-axis sensitivity. However with careful selection of transducers, these uncertainties should also be minimized (i.e., held to about one percent). Since the sensitivity of accelerometers is usually determined by laboratory calibration, this should also be well controlled. Although alignment of the sense axes can also produce errors, this may also be controlled by proper design and inspection procedures.

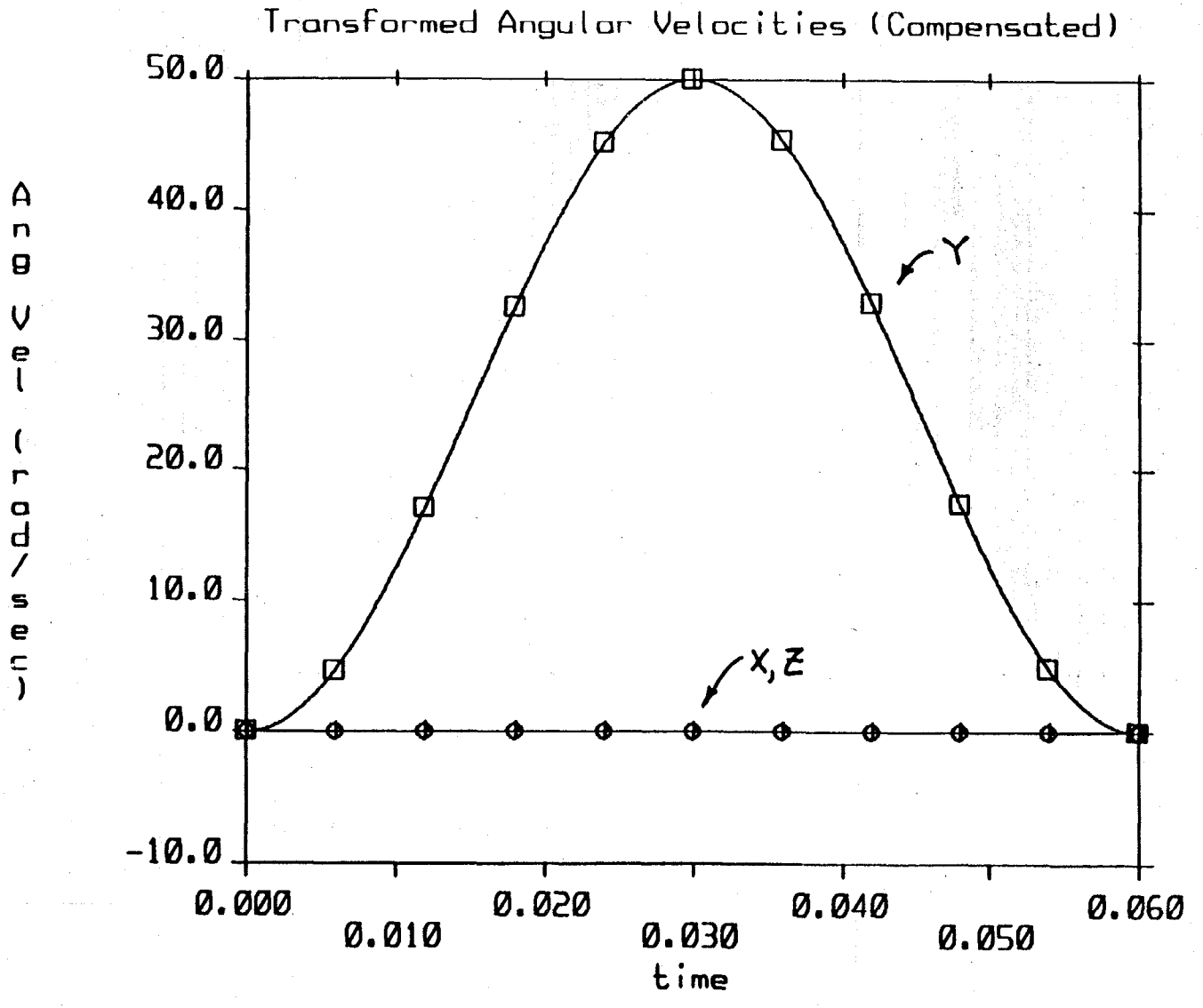


FIGURE 4-38. TRANSFORMED ANGULAR VELOCITIES (COMPENSATED FOR NON-COPLANAR GEOMETRY)

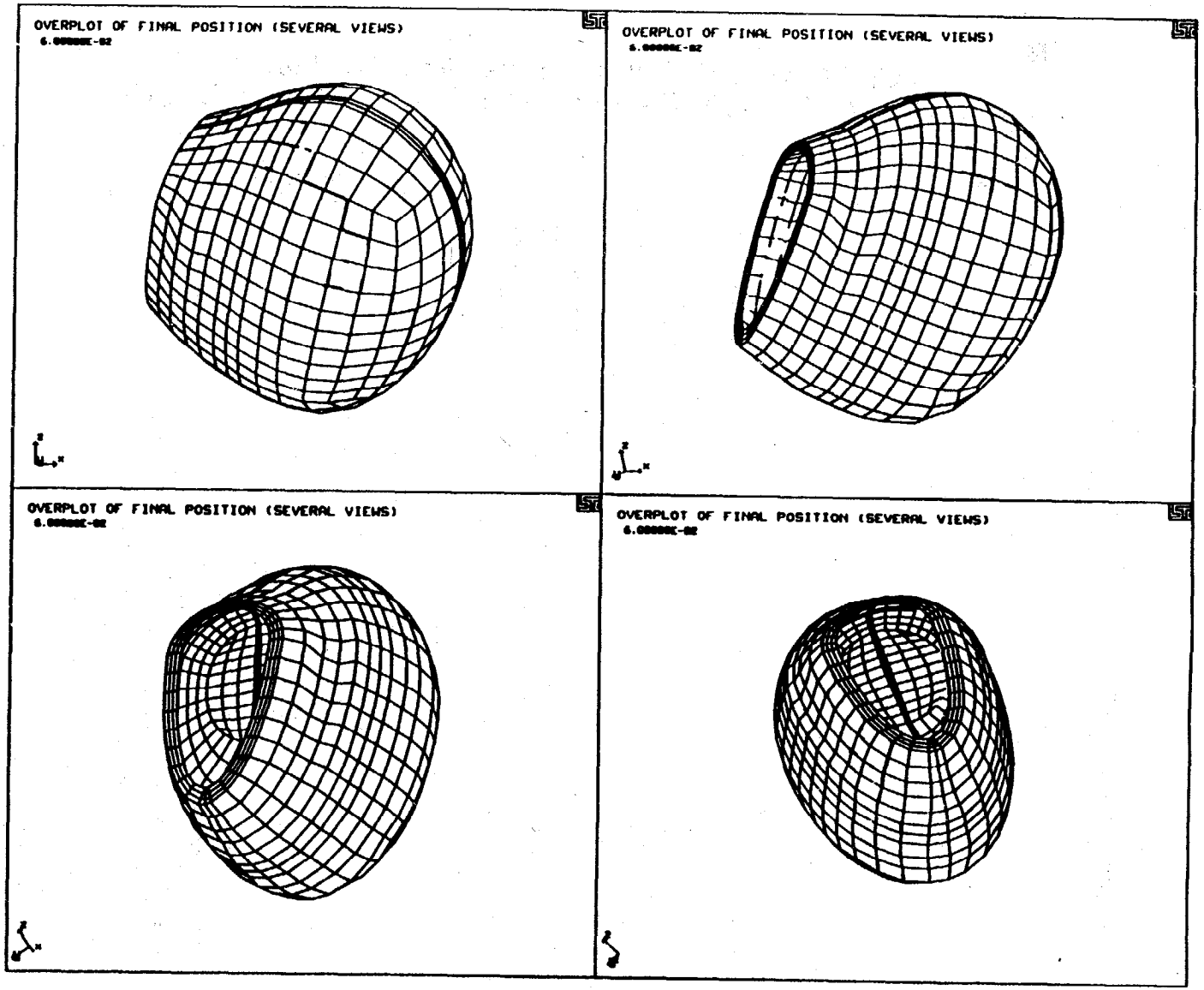


FIGURE 4-39. COMPARISON OF HEADPART LOCATION WITH CORRECT FINAL POSITION (COMPENSATED FOR NON-COPLANAR GEOMETRY)

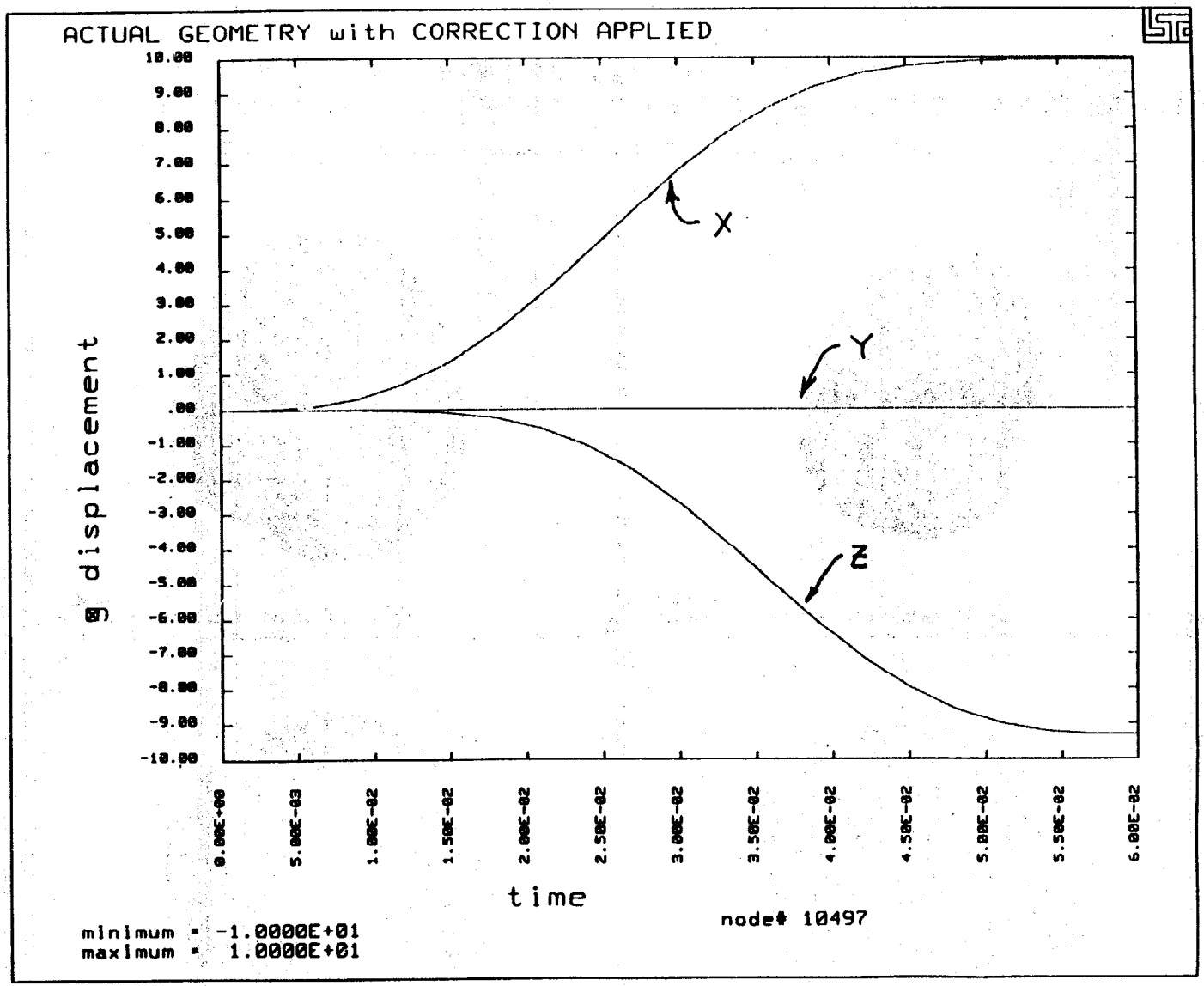


FIGURE 4-40. X, Y, AND Z TRANSLATIONAL DISPLACEMENTS AT CG (COMPENSATED FOR NON-COPLANAR GEOMETRY)

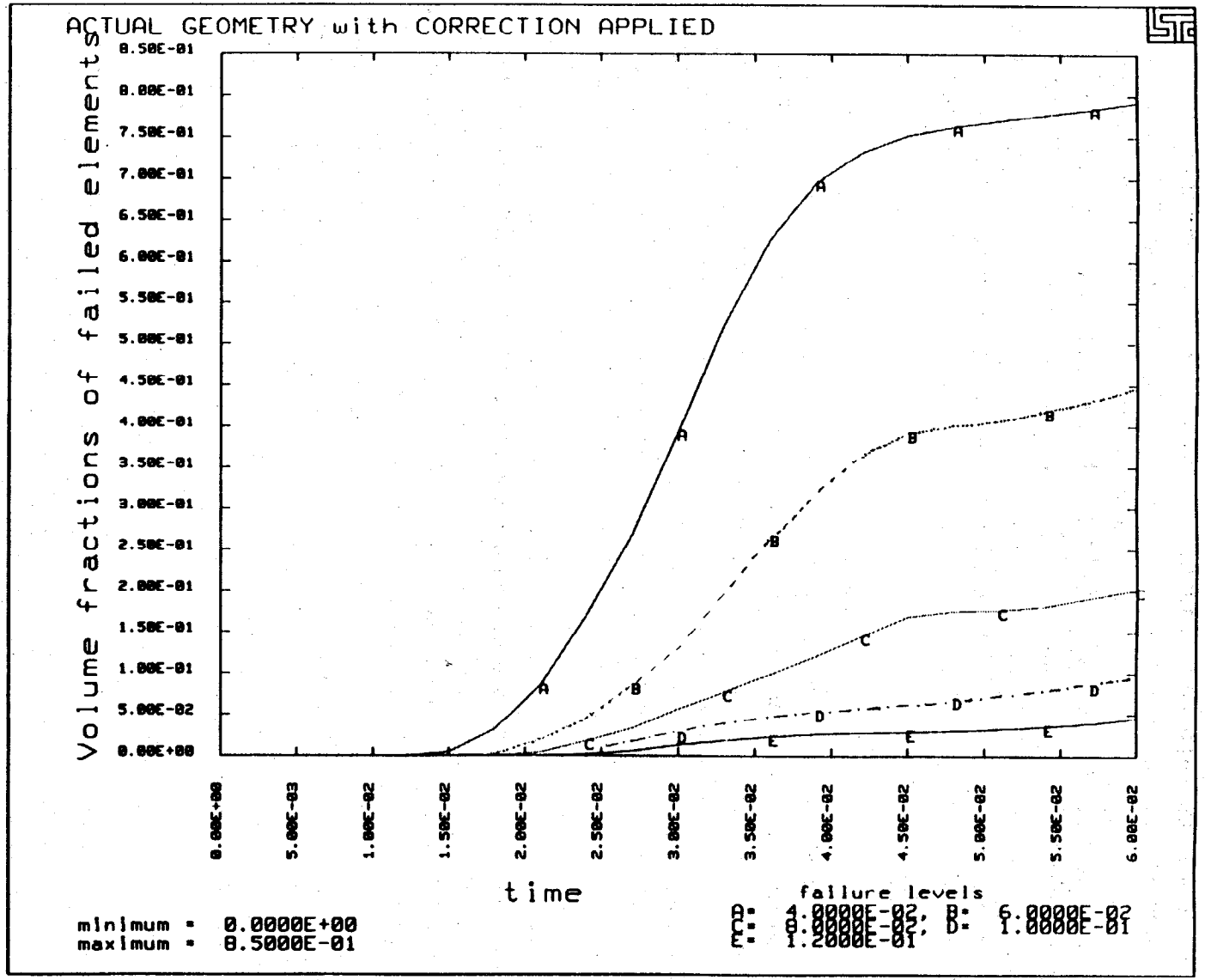


FIGURE 4-41. VOLUME FRACTION OF FAILED ELEMENTS (COMPENSATED FOR NON-COPLANAR GEOMETRY)

4.5.7 Effects of Uncompensated Transducer Error (1% Cross-Axis Sensitivity at CG)

To provide an indication of the relative effects that transducer uncertainties could have on kinematic response and strain computations, accelerometer data was generated to simulate a one percent cross-axis sensitivity error. To simplify the computations, the one percent cross-axis sensitivity was applied to accelerometers at the CG only. Nominal values of computed accelerations were multiplied by 0.01 times the vector sum of accelerations in the orthogonal plane at the CG. Although this is a preliminary evaluation of transducer uncertainties and does not include errors associated with accelerometers on the arms, accelerometers at the CG are used in all computations for body angular accelerations and should provide an indication of errors resulting from accelerometer cross-level sensitivity.

For this condition, Figure 4-42 illustrates transformed angular velocities and Figure 4-43 describes the kinematic response at the CG. The final position of the headpart overlaid with the correct final position is shown in Figure 4-44 for several views. From these results, some minor variations in kinematics and final position are apparent. From Figure 4-42, the Y axis angular velocity is almost exactly reproduced, with errors in X and Z axis angular velocities of about 0.5 and 0.8 rad/sec (1% and 1.6%), respectively. The maximum X translational displacement of 10.050 inches is overproduced by 0.050 inches (0.5%) and minor variations are also present in Y and Z translational displacements.

Failed volume data is shown in Figure 4-45 for this condition, and comparing this with the data of Figure 4-24, it can be seen that the results are essentially identical. These results suggest that uncertainties associated with the uncompensated non-coplanar geometry condition are more significant than variations associated with transducer effects such as cross-axis sensitivity. It should be noted however, that the calibration procedure described in [4] contains additional terms (ε_{ijk}) for compensating transducer measurement errors such as cross-axis sensitivity, should these be known for the specific accelerometer at each measurement location. Since the simulated 1% cross axis sensitivity does not have a significant effect; and since specific transducer sensitivity data is not generally available, this step has not been implemented.

4.6 MAGNETOHYDRODYNAMIC (MHD) SENSORS

Relatively compact angular rate sensors using magnetohydrodynamic (MHD) principles have recently been developed for the Department of Transportation for direct measurement of dummy headpart angular velocities. MHD packages use three angular rate sensors, each oriented along a principal body axis to provide an output proportional to angular velocity, with three translational accelerometers located near the CG and can

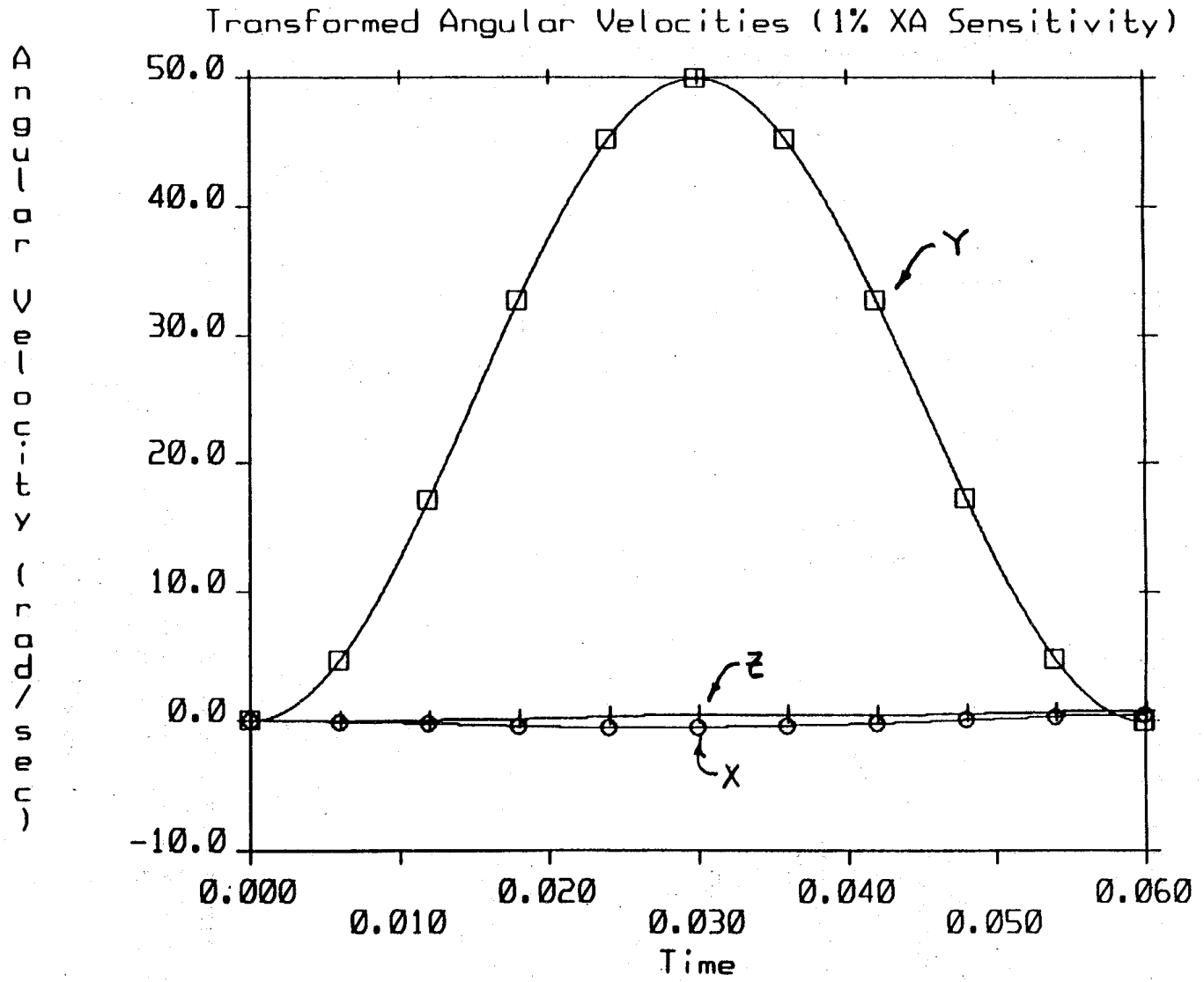


FIGURE 4-42. TRANSFORMED ANGULAR VELOCITIES (COMPENSATED NON-COPLANAR GEOMETRY WITH 1% C-A ERROR)

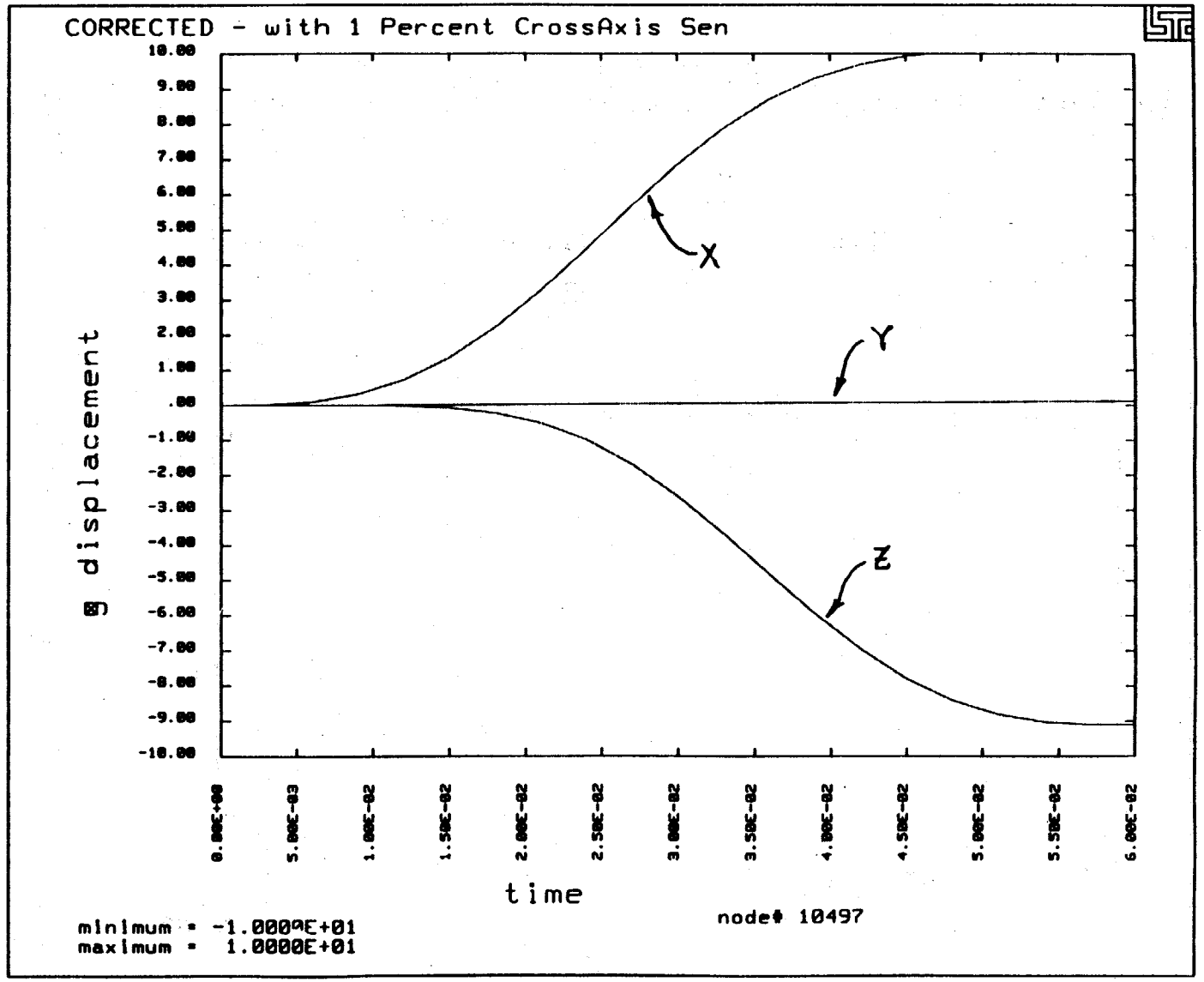


FIGURE 4-43. KINEMATIC RESPONSE AT THE CG (COMPENSATED NON-COPLANAR GEOMETRY WITH 1% C-A ERROR)

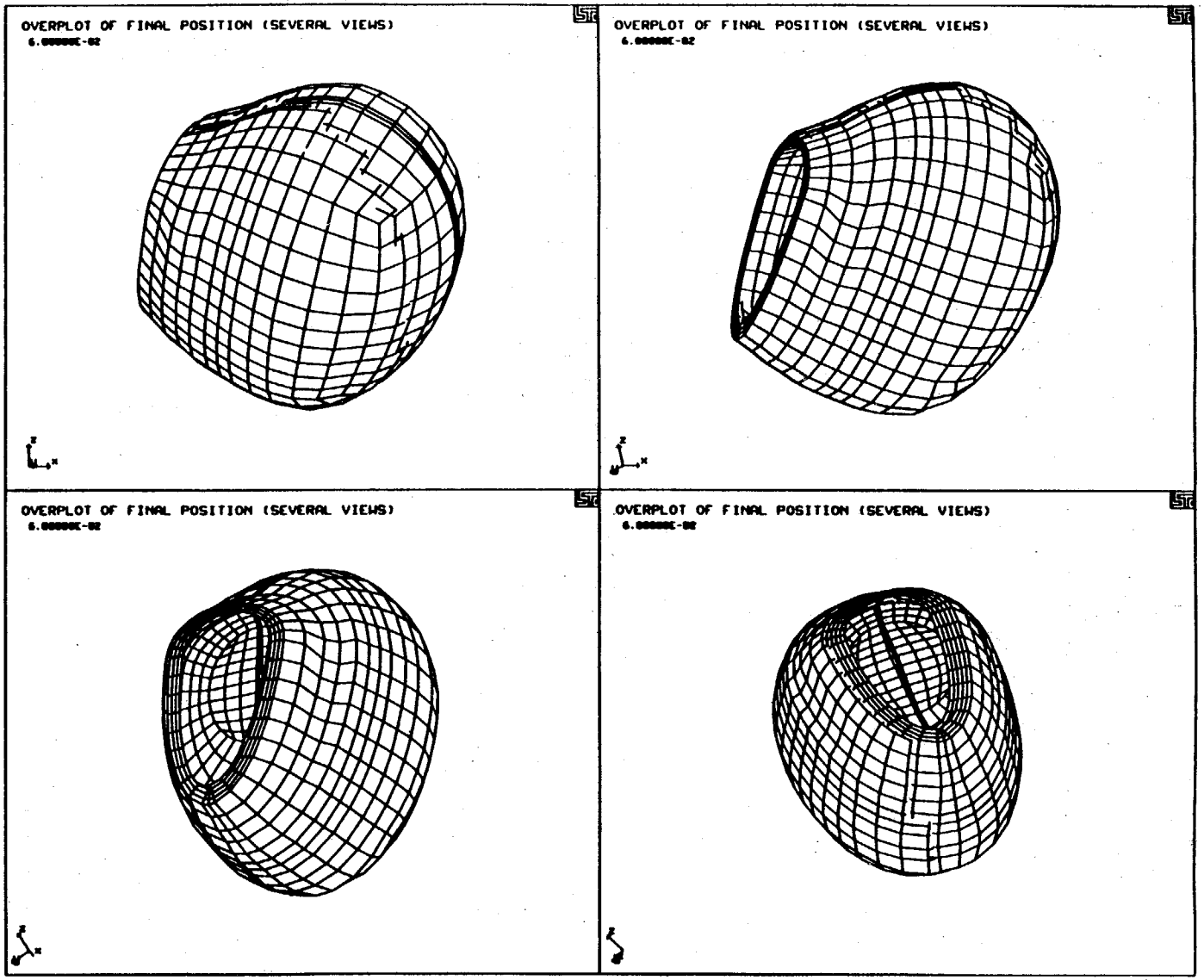


FIGURE 4-44. FINAL POSITION OF THE HEADPART OVERLAID WITH CORRECT FINAL POSITION (COMPENSATED NON-COPLANAR GEOMETRY WITH 1% CROSS-AXIS SENSITIVITY)

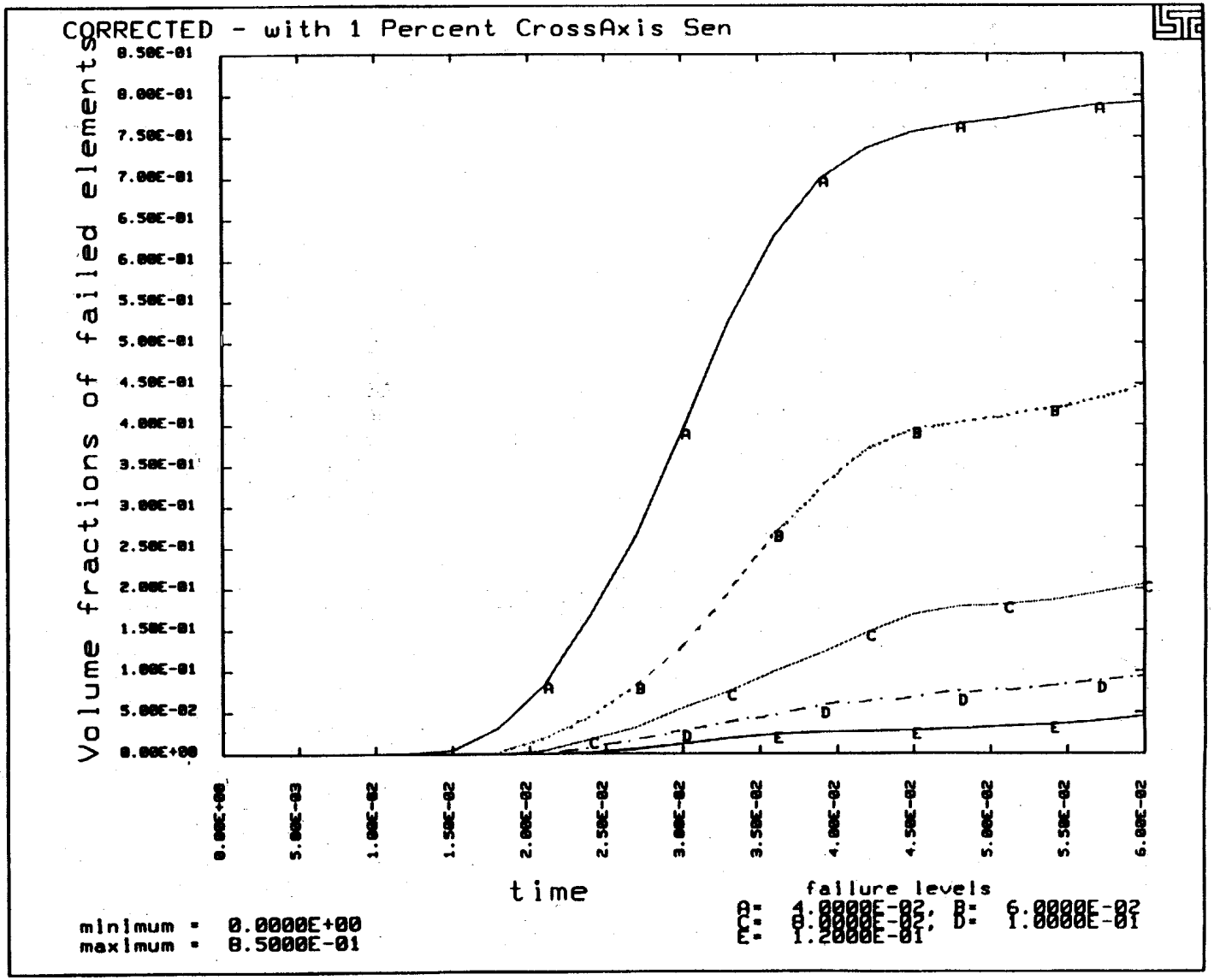


FIGURE 4-45. FAILED VOLUME DATA (COMPENSATED NON-COPLANAR GEOMETRY WITH 1% C-A ERROR)

also provide generalized six-degree-of-freedom kinematic response measurements. Because of the small MHD package size, it appears that careful axial alignment of MHD transducers with headpart body axes may be particularly important. Principles of operation and typical configurations of MHD sensor packages are described in [9].

Since these transducers have an output signal proportional to angular velocity, the integration of angular accelerations is not required, and angular velocities may be taken directly from the MHD sensors. However, the orientation of the headpart relative to fixed coordinates must also be determined at each time step and accelerometer measurements made at CG locations must also be compensated and transformed to inertial coordinates as was done for the 3-2-2-2 NAP array, to provide a generalized, six-degree-of-freedom description of velocities at the CG.

4.6.1 Magnetohydrodynamic (MHD) Sensor Data and the Transformation Process

Although MHD data could be used to replace off-axis accelerometers in the NAP array and in the transformation process (by substituting computed values of Bomeg*x*, *y* and *z* with the direct MHD measurements and removing computations associated with compensation of angular kinematics), there are advantages to making angular velocity measurements with both MHD and NAP packages. Since the MHD sensors can be packaged with accelerometers located at the CG (displacing MHD devices to off-CG locations), providing data from both measurement systems would require a total of 12 data channels, only three more than currently used for the NAP array. Non-coplanar geometry compensation coefficients (Appendix A) are easily re-evaluated if modifications to the NAP array geometry were necessary to accommodate a combined NAP-MHD measurement system.

A combined NAP-MHD system would provide a redundant measurement of angular kinematics and provide additional data for studying the reliability and relative strengths of each measurement system. Independent of the measurement system used, the transformation process and compensation algorithm described previously is required and should be implemented for accurate computation of headpart kinematics with respect to inertial coordinates.

APPENDIX A

IMPLEMENTATION OF COMPENSATION ALGORITHM FOR 3-2-2-2 NON-COPLANAR NAP GEOMETRY

In the transformation process, body angular accelerations are initially estimated by computing rotational accelerations in each orthogonal plane, assuming a nominal or ideal location of accelerometers within the array (i.e. two or more accelerometer sense axes located at the same physical location). These angular accelerations are then integrated to estimate body angular velocities which are used in turn, to track the body coordinate system relative to a fixed coordinate system using an updated direction cosine matrix. Transformation error buildup is minimized by limiting angular displacements to 1 milliradian or less, during each integration interval.

To begin the process, angular accelerations are estimated using two pairs of translational accelerometers, each separated by a radial arm distance, about each body axis. These angular acceleration estimates, based on the ideal location of accelerometers at the CG and on each arm of the array are:

$$\dot{\Omega}_x = ((bacc4 - bacc3) / ry + (bacc2 - bacc9) / rz) / 2$$

$$\dot{\Omega}_y = ((bacc8 - bacc1) / rz + (bacc3 - bacc6) / rx) / 2 \qquad \text{(A-1)}$$

$$\dot{\Omega}_z = ((bacc7 - bacc2) / rx + (bacc1 - bacc5) / ry) / 2$$

where

*bacc*1 through *bacc*9 are measured accelerometer outputs (i.e. NAP data - refer to Table 2-1 and Figure 4-25) and $\dot{\Omega}_x, \dot{\Omega}_y, \dot{\Omega}_z$ are angular acceleration estimates based on coplanar geometry

From Appendix A1-1 of the calibration procedure described in Reference 4, the following equations were developed to relate the above estimates of angular acceleration to *actual* angular accelerations, and error terms resulting from the combined effects of (a) non-coplanar location of accelerometers in the NAP array and/or (b) transducer measurement errors.

$$\dot{\Omega}_x = \dot{\omega}_x + A_0 + A_1\dot{\omega}_x + A_2\dot{\omega}_y + A_3\dot{\omega}_z + A_4\omega_x^2 + A_5\omega_y^2 + A_6\omega_z^2 + A_7\omega_x\omega_y + A_8\omega_x\omega_z + A_9\omega_y\omega_z + A_{10}\ddot{X} + A_{11}\ddot{Y} + A_{12}\ddot{Z}$$

$$\dot{\Omega}_y = \dot{\omega}_y + B_0 + B_1\dot{\omega}_x + B_2\dot{\omega}_y + B_3\dot{\omega}_z + B_4\omega_x^2 + B_5\omega_y^2 + B_6\omega_z^2 + B_7\omega_x\omega_y + B_8\omega_x\omega_z + B_9\omega_y\omega_z + B_{10}\ddot{X} + B_{11}\ddot{Y} + B_{12}\ddot{Z} \quad \text{(A-2)}$$

$$\dot{\Omega}_z = \dot{\omega}_z + C_0 + C_1\dot{\omega}_x + C_2\dot{\omega}_y + C_3\dot{\omega}_z + C_4\omega_x^2 + C_5\omega_y^2 + C_6\omega_z^2 + C_7\omega_x\omega_y + C_8\omega_x\omega_z + C_9\omega_y\omega_z + C_{10}\ddot{X} + C_{11}\ddot{Y} + C_{12}\ddot{Z}$$

where: A_i, B_i, C_i are geometry and/or transducer error coefficients
$\omega_x, \omega_y, \omega_z$ are actual body angular velocity components,
$\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z$ are actual body angular acceleration components, and
$\ddot{X}, \ddot{Y}, \ddot{Z}$ are measured body acceleration components at CG

The terms of Equation A-2 may be regrouped as:

$$\underset{\sim}{a}\,\underset{\sim}{\dot{\omega}} = \underset{\sim}{\dot{\Omega}} - \underset{\sim}{f}(\underset{\sim}{\omega}, \underset{\sim}{\ddot{X}}) \quad \text{(A-3)}$$

where

$$\underset{\sim}{a} = \begin{bmatrix} 1 + A_1 & A_2 & A_3 \\ B_1 & 1 + B_2 & B_3 \\ C_1 & C_2 & 1 + C_3 \end{bmatrix} \quad \text{(A-4a)}$$

and considering A_0, B_0, C_0 are transducer error coefficients and zero if perfect transducers are assumed,

$$\underset{\sim}{f} = \begin{bmatrix} A_{10} & A_{11} & A_{12} \\ B_{10} & B_{11} & B_{12} \\ C_{10} & C_{11} & C_{12} \end{bmatrix} \begin{Bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{Bmatrix} + \begin{bmatrix} A_7 & A_8 & A_9 \\ B_7 & B_8 & B_9 \\ C_7 & C_8 & C_9 \end{bmatrix} \begin{Bmatrix} \omega_x\omega_y \\ \omega_x\omega_z \\ \omega_y\omega_z \end{Bmatrix} + \begin{bmatrix} A_4 & A_5 & A_6 \\ B_4 & B_5 & B_6 \\ C_4 & C_5 & C_6 \end{bmatrix} \begin{Bmatrix} \omega_x^2 \\ \omega_y^2 \\ \omega_z^2 \end{Bmatrix} \quad \text{(A-4b)}$$

These equations relate actual body angular accelerations to estimates of angular acceleration based coplanar geometry, and cross products of angular velocities and centripetal accelerations arising from non-coplanar location of accelerometers.

After evaluating geometry error coefficients, the actual angular accelerations are obtained from:

$$\underset{\sim}{\dot{\omega}} = \underset{\sim}{a}^{-1}[\underset{\sim}{\dot{\Omega}} - \underset{\sim}{f}(\underset{\sim}{\omega}, \underset{\sim}{\ddot{X}})] \quad \text{(A-5)}$$

Evaluation of Geometry Compensation Coefficients:

Accelerometer location errors are defined in Reference 4 as δ_{ijk} and are functions of accelerometer location, sense axis direction, and variation from ideal geometry.

i indicates accelerometer location (0,1,2 or 3 for CG, x, y or z arm),
j indicates accelerometer measurement direction (x, y, or z)
k direction of variation from ideal coplanar geometry
(1,2,3 corresponding to x, y, z direction).

An additional error source, ε_{ijk} is associated with transducer errors. Null values have been assigned to these terms since perfect transducers are assumed in the following discussion. However, with knowledge of measurement error terms such as cross-axis sensitivity at specific transducer locations, a similar procedure could easily be applied to compensate for these effects.

Geometry compensation coefficients δ_{ijk}, for the non-coplanar geometry of Table 4-2 are indicated below for each accelerometer location:
At the cg:

(Bacc1)	(Bacc2)	(Bacc3)
d0x1 = 0.203	d0y1 = 0.0	d0z1 = 0.0
d0x2 = 0.0	d0y2 = 0.203	d0z2 = 0.0
d0x3 = 0.0	d0y3 = 0.0	d0z3 = 0.203

On the x arm (Using location of bacc6 for reference, rx = 2.403)

(Bacc7)	(Bacc6)
d1y1 = -.203	d1z1 = 0.0
d1y2 = 0.0	d1z2 = 0.0
d1y3 = 0.203	d1z3 = 0.0

On the y arm (Using location of bacc4 for reference, ry = 2.103)

(Bacc5)	(Bacc4)
d2x1 = 0.0	d2z1 = 0.0
d2x2 = -0.203	d2z2 = 0.0
d2x3 = .203	d2z3 = 0.0

On the z arm (Using z axis as reference position, rz = 3.20)

(Bacc8)	(Bacc9)
d3x1 = 0.203	d3y1 = 0.0
d3x2 = 0.0	d3y2 = 0.203
d3x3 = 0.0	d3y3 = 0.0

Evaluating non-zero compensation coefficients A_i, B_i, C_i for the above geometry yields the following:

A_4 = -(d2z3 - d0z3)/(2.*ry) + (d3y2 - d0y2)/(2.*rz) = 0.203/(2.*ry)

A_5 = -(d2z3 - d0z3)/(2.*ry) = 0.203/(2.*ry)

B_4 = (d1z3 - d0z3)/(2.*rx) = -.0203/(2.*rx)

B_5 = -(d3x1 - d0x1)/(2.*rz) + (d1z3 - d0z3)/(2.*rx) = -0.203/(2.*rx)

C_1 = -(d1y3 - d0y3)/(2.*rx) = -0.203/(2.*rx)

C_2* = -(d2x3 - d0x3)/(2.*ry) = -2.03/(2.*rz)

C_3 = (d1y1 - d0y1)/(2.*rx) + (d2x2 - d0x2)/(2.*ry)
= -0.203/(2.*rx) - 0.203/(2.*ry)

C_4 = -(d1y2 - d0y2)/(2.*rx) = 0.203/(2.*rx)

C_5 = (d2x1 - d0x1)/(2.*ry) = -0.203/(2.*ry)

C_6 = -(d1y2 - d0y2)/(2.*rx) + (d2x1 - d0x1)/(2.*ry)
= 0.203/(2.*rx) - 0.203/(2.*ry)

C_7* = (d1y1 - d0y1)/(2.*rx) - (d2x2 - d0x2)/(2.*ry)
= -0.203/(2.*rx) + 0.203/(2.*ry)

C_8 = -(d2x3 - d0x3)/(2.*ry) = -0.203/(2.*ry)

C_9 = (d1y3 - d0y3)/(2.*rx) = 0.203/(2.*rx)

For these coefficients, Equation A-2 reduces to the following:

$$\dot{\Omega}_x = \dot{\omega}_x + A_4 \omega_y^2 + A_5 \omega_z^2$$

$$\dot{\Omega}_y = \dot{\omega}_y + B_4 \omega_x^2 + B_5 \omega_y^2 \qquad \text{(A-6)}$$

$$\dot{\Omega}_z = (1 + C_3)\dot{\omega}_z + C_1\dot{\omega}_x + C_2\dot{\omega}_y + C_4\omega_x^2 + C_5\omega_y^2 + C_6\omega_z^2 + C_7\omega_x\omega_y + C_8\omega_x\omega_z + C_9\omega_y\omega_z$$

and the matrices $\underset{\sim}{a}$ and $\underset{\sim}{f}(\underset{\sim}{\omega}, \underset{\sim}{\ddot{X}})$ become:

$$\underset{\sim}{a} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ C_1 & C_2 & 1+C_3 \end{bmatrix} \tag{A-7}$$

$$\underset{\sim}{f}(\underset{\sim}{\omega}, \underset{\sim}{\ddot{X}}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ C_7 & C_8 & C_9 \end{bmatrix} \begin{Bmatrix} \omega_x\omega_y \\ \omega_x\omega_z \\ \omega_y\omega_z \end{Bmatrix} + \begin{bmatrix} A_4 & A_5 & 0 \\ B_4 & B_5 & 0 \\ C_4 & C_5 & C_6 \end{bmatrix} \begin{Bmatrix} \omega_x^2 \\ \omega_y^2 \\ \omega_z^2 \end{Bmatrix} \tag{A-8}$$

From above, $\underset{\sim}{f}(\underset{\sim}{\omega}, \underset{\sim}{\ddot{X}})$ becomes

$$f_1 = A_4\omega_x^2 + A_5\omega_y^2$$

$$f_2 = B_4\omega_x^2 + B_5\omega_y^2 \tag{A-9}$$

$$f_3 = C_4\omega_x^2 + C_5\omega_y^2 + C_6\omega_z^2 + C_7\omega_x\omega_y + C_8\omega_x\omega_z + C_9\omega_y\omega_z$$

Substituting into equation A-5, actual angular accelerations are obtained from:

$$\underset{\sim}{\dot{\omega}} = \underset{\sim}{a}^{-1}\left\{\dot{\Omega}_i - f_i\right\} \tag{A-10}$$

Where

$$\underset{\sim}{a}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -C_1/(1+C_3) & -C_2/(1+C_3) & 1/(1+C_3) \end{bmatrix} \tag{A-11}$$

As indicated by the number of non-zero coefficients, the accelerometer locations indicated in Table 4-2 are not the most advantageous selections, but they are general and provide a good test of the compensation algorithm.

This compensation algorithm has been integrated into the coordinate transformation procedure to provide improved estimates of body angular acceleration, by compensating for centripetal accelerations and cross products of angular velocity arising from generalized non-coplanar location of accelerometers in the NAP array.

Denton 3-2-2-2 NAP Configuration:

The Denton 3-2-2-2 geometry is non-coplanar, but uses judicious selection of accelerometer locations to minimize cross-products of angular velocity and centripetal accelerations. This is apparent in the computation of the compensation coefficients A_i, B_i, C_i. Location of accelerometers in the Denton 3-2-2-2 array are indicated below, based on drawing no. B-1846, provided by Robert A. Denton Inc.

POSITION OF ACCELEROMETERS IN DENTON 3-2-2-2 NINE ACCELEROMETER PACKAGE ARRAY

Accel Location	Sense Axis	Accel Label	X axis Location	Y axis Location	Z axis Location
CG	X	Bacc1	0.203	0.000	0.000
CG	Y	Bacc2	0.000	0.203	0.000
CG	Z	Bacc3	0.000	0.000	0.203
Y Arm	Z	Bacc4	0.000	1.900	0.203
Y Arm	X	Bacc5	0.203	1.900	0.000
X Arm	Z	Bacc6	2.200	0.000	0.203
X Arm	Y	Bacc7	2.200	0.203	0.000
Z Arm	X	Bacc8	0.203	0.000	3.200
Z Arm	Y	Bacc9	0.000	0.203	3.200

Geometry variation coefficients δ_{ijk} defining differences between the non-coplanar array geometry and ideal coplanar geometry are evaluated at each transducer location using the appropriate geometry data. For the Denton 3-2-2-2 NAP package, variations from coplanar geometry at each location (i.e. Bacc*i* *i*=1 to 9) are indicated below:

At the cg:

(Bacc1)	(Bacc2)	(Bacc3)
d0x1 = 0.203	d0y1 = 0.0	d0z1 = 0.0
d0x2 = 0.0	d0y2 = 0.203	d0z2 = 0.0
d0x3 = 0.0	d0y3 = 0.0	d0z3 = 0.203

On the x arm (Using location of bacc6 for reference, rx = 2.200)

(Bacc7)	(Bacc6)
d1y1 = 0.0	d1z1 = 0.0
d1y2 = 0.203	d1z2 = 0.0
d1y3 = 0.0	d1z3 = 0.203

On the y arm (Using location of bacc4 for reference, ry = 1.900)

(Bacc5)	(Bacc4)
d2x1 = 0.203	d2z1 = 0.0
d2x2 = 0.0	d2z2 = 0.0
d2x3 = 0.0	d2z3 = 0.203

On the z arm (Using z axis as reference position, rz = 3.20)

(Bacc8)	(Bacc9)
d3x1 = 0.203	d3y1 = 0.0
d3x2 = 0.0	d3y2 = 0.203
d3x3 = 0.0	d3y3 = 0.0

This geometry yields a null set of compensation coefficients A_i, B_i, C_i, indicating that for this configuration, angular accelerations may be computed exactly, and compensation is not required, other than for accelerations measured at the cg as discussed below. As noted in [4] this procedure may also be used to compensate for transducer measurement errors such as cross axis sensitivity.

Compensation of Acceleration Measurements at CG:

In addition to the above adjustments, accelerometers located at the Center of Gravity (CG) must also be adjusted since the offset distances from the CG will result in centripetal accelerations which will reduce the value of acceleration made at the exact CG location. From Reference 4, the relationship between actual and measured accelerations at the CG location is:

$$\begin{aligned} \ddot{x} &= bacc1 + \delta_x * (\omega_y^2 + \omega_z^2) \\ \ddot{y} &= bacc2 + \delta_y * (\omega_x^2 + \omega_z^2) \\ \ddot{z} &= bacc3 + \delta_z * (\omega_y^2 + \omega_x^2) \end{aligned} \qquad \text{(A-12)}$$

Where $\ddot{x}, \ddot{y}, \ddot{z}$ are actual body accelerations at the CG and
$\delta_x, \delta_y, \delta_z$ are transducer offset distances along the x, y and z arms.

For the geometry of Table 4-2 and for the Denton 3-2-2-2 array geometry, $\delta_x = \delta_y = \delta_z = 0.203$ inches.

Solving the above equations during the transformation process provides improved estimates of angular and translational kinematics by compensating for cross products of angular velocity and centripetal accelerations arising from the non-coplanar location of accelerometers in the NAP array.

Errata: From an inspection of the equations in Appendix A1-1 of the Calibration Report (Reference 4), errors were found in expressions for the following coefficients:

As described in Reference 4:

```
c2 = -(d2x3 - d0x3)/(2.*rz)
c7 =  (d1y1 - d0y1)/(2.*rx) - (d2y2 - d0y2)/(2.*ry)
```

Corrected expressions used in the compensation algorithm above:

```
c2 = -(d2x3 - d0x3)/(2.*ry)
c7 =  (d1y1 - d0y1)/(2.*rx) - (d2x2 - d0x2)/(2.*ry)
```

APPENDIX B

PROGRAM LISTING FOR NINE-ACCELEROMETER-PACKAGE (NAP) COORDINATE TRANSFORMATION SOFTWARE

```
c=============================================================
c
c     PROGRAM LISTING FOR NINE-ACCELEROMETER-PACKAGE (NAP)
c           COORDINATE TRANSFORMATION SOFTWARE
c
c=============================================================
      !
      PROGRAM LC_COMP
      !
      ! Includes compensation for non-coplanar NAP geometry
      !
      !
      ! Programmed by F. DiMasi and H. Weinstock DTS-76
      ! Volpe National Transportation Systems Center,
      ! 55 Broadway
      ! Cambridge Ma.  02142
      ! (Tel 617-494-2277)
      !
      !
      ! ============================================
      ! Notes:
      !
      ! Units are  IN-LB-SEC  for compatibility with anatomic brain model.
      !
      ! Input and output file files in the following code are in the form of
      !  NHTSA UDS formatted files.  These sections should be modified
      !  to read data in other formats.
      !
      ! NAP DATA MUST BE READ IN A SPECIFIC ORDER such that
      !  the input accelerometer data corresponds to the appropriate position
      !  within the nine accelerometer package.
      !
```

```
!   (Alternatively, data may be keyed to positional information, if available
!   with input data - e.g., sensor attachment point and sense axis
!   for NHTSA UDS data files.)
!
! Read 9 Accelerometer files, i = 1 to 9, ordered as follows
!
! ================================================================
!
! Input File No.      Arm    Axis
!
!        1            Xcg    X
!        2            Ycg    Y
!        3            Zcg    Z
!        4            Yarm   Z
!        5            Yarm   X
!        6            Xarm   Z
!        7            Xarm   Y
!        8            Zarm   X
!        9            Zarm   Y
!
! ================================================================
!
! The following load curves apply to the body center-of-gravity and
!  are written to the output file lc_file.dat
!
!  Translational velocity along fixed X coordinate
!  Translational velocity along fixed Y coordinate
!  Translational velocity along fixed Z coordinate
!  Angular velocity along fixed X coordinate
!  Angular velocity along fixed Y coordinate
!  Angular velocity along fixed Z coordinate
!
! Modified on 6/28/94  to add:
!   (a) compensation algorithm and
!   (b) initial angular orientation
!
! ================================================================
!
! Inputs/variables:
!
!  bacc(i)  i=1 to 9 ;  translational accelerations as measured with
!                       a 3-2-2-2 NAP in body coordinates
!
```

```
! omegdn  n=1 to 3 ;  estimates of averaged x, y, z angular accel in body
!   coordinates (computed from baccn, n= 4 thru 9)
!
! Outputs:
!
!  faccn    n = x, y, & z  ;  Trans. accel, fixed coordinates
!
!  fveln    n = x, y, & z ;   Trans. velocity, fixed coordinates
!
!  fomegnn  nn = xx, yy, & zz ;  Angular vel's about fixed axes
!
!  bomegnn  nn = xx, yy & zz ;   Angular vel's about body axes
!
!  bomegdn  n = x, y & x ;   Angular accel's about body axes
!
!  angn             n = 1 to 3 ;  Angle between body and fixed axes
!
! ================================================
! Begin reading NAP data - Structure for NHTSA UDS files
! ================================================
!
! Include the UDS structure include file. This include file
! also includes other files that define platform characteristics
! and array limits
!
include 'uds_str.fh'
!
! Declare program defined bounds for local arrays.
!
integer bot,top
parameter(bot=-1000,top=10000)
!
! Declare the error status flag
!
integer ios
!
! Declare variables to hold the lower and upper array bounds
!
integer         lb, ub
!
! Miscellaneous counter variables
!
```

```
      integer i,j
      !
      ! Logical variable to determine if file exists.
      !
      logical fexists
      !
      ! Declare a UDS control structure, and a template UDS structure
      ! (object) to be used to read in signal data - a single UDS
      ! structure is used for all I/O, with signal data being stored
      ! in local arrays rather than multiple UDS objects
      !
      record/uds_control/ uds_c

      record/uds/ template
      !
      ! Declare and dimensions local work arrays
      !
      real*4 omeg(3,3),timeout(5000)
      !
      ! Declare local DEL - time increment
      !
      real*4 del
      !
      ! Declare a bacc_struct.
      !
      structure /bacc_struct/
            real*4 y(bot:top)
      end structure

      record/bacc_struct/ bacc(9)
      !
      ! Declare local working arrays to be used for computations
      !
      real*4 bomegxx(bot:top),bomegyy(bot:top),bomegzz(bot:top),
     *       bomegdx(bot:top),bomegdy(bot:top),bomegdz(bot:top),
     *       faccx(bot:top),faccy(bot:top),faccz(bot:top),
     *       fvelx(bot:top),fvely(bot:top),fvelz(bot:top),
     *       fomegdx(bot:top),fomegdy(bot:top),fomegdz(bot:top),
     *       fomegx(bot:top),fomegy(bot:top),fomegz(bot:top),
     *       ang1(bot:top),ang2(bot:top),ang3(bot:top),
     *       alpha(bot:top),beta(bot:top),gamma(bot:top),
     *       Q(3),R(3),AI(3,3)
```

```
!
double precision T(3,3), TDT(3,3)
!
! Declare arrays of character strings to hold input and output
! file names
!
character*20 infile(9),outfile(21)
!
! Declare arrays to hold values used in initial input routines
!
character*3  fpos(9)/'1st','2nd','3rd',
*                    '4th','5th','6th',
*                    '7th','8th','9th'/
character*7  ftag(9)/'  Xcg  ',
*                    '  Ycg  ',
*                    '  Zcg  ',
*                    'Yarm Zg',
*                    'Yarm Xg',
*                    'Xarm Zg',
*                    'Xarm Yg',
*                    'Zarm Xg',
*                    'Zarm Yg'/
!
! Declare the testno variable for holding the test number
!
character testno*4
!
! Declare external functions
!
integer len_trim
external len_trim
!
! Initialize UDS control structure
!
uds_c.dimsys = 'ENG'
uds_c.chanform = 'Y'
uds_c.fill_x = .false.
uds_c.no_msg = .false.
!
! Define initial lower and upper bounds as size of local arrays
!
lb = bot
```

```
      ub = top
      !
      ios = 0

800   continue

      write (6,801)
801   format (//3X,'*** Transform NAP Data From Body',1x,
     *  'to Inertial Coordinates ***'//)

      write(6, 6013)
      read(5, *) phi, theta, psi

6013 FORMAT(/, 2X,
     * '- PHI, THETA, PSI (rad): angle of rotation of accelerometer', /,
     * '  system about, (1) x-axis, (2) y-axis, and (3) z-axis', /,
     * 2X, '...PHI/(E,F), THETA/(E,F), PSI/(E,F): ')
      !
      ! Begin section to retrieve filenames
      !
900   continue
      !
      ! Declare common format statements used in loop
      !
901   format(/4x,'Enter ',a,' UDS file in NAP array (',a,')',
     *       /4x,'Control-Z to Stop >> ',$)
902   format(/4x,'*** ERROR: File does not exist : ',a)
      !
      ! Loop and read filenames from user
      !
      do i=1,9
      !
      ! Write prompt using appropriate values from 'description' arrays
      !
910         write(*,901) fpos(i),ftag(i)
      !
      ! Read the filename from the user
      !
            read(*,'(a)',end=990) infile(i)
      !
      ! Use the inquire() call to see if the file exists
      !
```

```
        fexists = .false.
        inquire(file=infile(i),exist=fexists)
!
! Warn user for non-existing file and re-prompt
!
        if (.not.fexists) then
            write(*,902)
*           infile(i)(:len_trim(infile(i)))
            goto 910
        endif
    enddo
!
! Read UDS, verify existence and set lower and upper bounds
!
    i = 1
!
! Read specification data only
!
    uds_c.specs_only = .true.
!
! Loop to read additional files
!
    do while ((i.le.9).and.(ios.eq.0))
!
! Read UDS file into UDS 'object', template, per uds_c
!
        call uds_read(infile(i),template,uds_c,ios)
!
! If no error, define number of points in array as
! npts = (ub-lb)+1
!
        if (ios.eq.0) then
            write(*,'(/6x,a,a)')
*   'Reading file: ',infile(i)(:len_trim(infile(i)))
            lb = max(lb,template.nfp)
            ub = min(ub,template.nlp)
        endif
!
! Increment counter for next UDS file
!
        i = i + 1
    enddo
```

```
!
! For file read error here, start over and re-prompt
! for files
!
if (ios.ne.0) then
        goto 800
endif
!
! Read each file and load signal data from UDS
! 'object' into local array
!
uds_c.specs_only = .false.
i = 1
do while ((i.le.9).and.(ios.eq.0))
        call uds_read(infile(i),template,uds_c,ios)
        call
*       copy_array(template.y,nfp_l,nlp_u,
*               bacc(i).y,bot,top,lb,ub)
        i = i + 1
enddo
!
! Finished reading UDS files
!
! Prepare UDS file as a template for writting output files
! here, modifying only parameters necessary for each file
!
uds_c.specs_only = .true.
call uds_read(infile(1),template,uds_c,ios)
!
! Set the NFP/NLP range appropriately
!
template.nfp = lb
template.nlp = ub
!
! Set the internal DEL to be the DEL of the template UDS file
!
del = template.del
!
! Set the test number
!
write(template.tstnam,'(a)') infile(1)(2:5)
write(testno,'(a)') infile(1)(2:5)
```

```
! ==========================================
! Finished reading UDS files and
! setting up output file templates
! ==========================================
! ==================
! Begin computations:
! ==================
! ============================================================
! Define nominal geometry for 3-2-2-2 NAP package to estimate
! angular accelerations and  subsequent compensation
! ============================================================
!
Gacc = 386.088
HGacc = Gacc*0.500
!
! ==========================================
!        Denton Nap Geometry Data:
! ==========================================
!
! Arm length along X, forward axis, taken to bacc6 (2.200 inches)
!
rx = 2.200
!
! Arm length along Y, lateral axis, taken to bacc4 (1.900 inches)
!
ry = 1.900
!
! Arm length along Z, vertical axis, taken to bacc8-9 (3.20 inches)
!
rz = 3.20
!
! ============================================
! Define error coefficients indicating variations from
!        coplanar geometry
! ============================================
!
! At the cg:
!  to Bacc1
d0x1 = 0.203
d0x2 = 0.0
d0x3 = 0.0

! to Bacc2
d0y1 = 0.0
```

```
d0y2 = 0.203
d0y3 = 0.0

! to Bacc3
d0z1 = 0.0
d0z2 = 0.0
d0z3 = 0.203

! On the x arm, using location of bacc6 for reference, rx = 2.200:
! to Bacc7
d1y1 = 0.0
d1y2 = 0.203
d1y3 = 0.0

! to Bacc6
d1z1 = 0.0
d1z2 = 0.0
d1z3 = 0.203

! On the y arm, using location of bacc4 for reference, ry = 1.900:
! to Bacc5
d2x1 = 0.203
d2x2 = 0.0
d2x3 = 0.0

! to Bacc4
d2z1 = 0.0
d2z2 = 0.0
d2z3 = 0.203

! On the z arm, using z axis as reference position, rz = 3.20:
! to Bacc8
d3x1 = 0.203
d3x2 = 0.0
d3x3 = 0.0

! to Bacc9
d3y1 = 0.0
d3y2 = 0.203
d3y3 = 0.0

! End Denton Geometry coefficients
! ================================
```

```
! ==========================================================
! Define nominal geometry for 3-2-2-2 NAP package to estimate
!       angular accelerations and  subsequent compensation
! (Assumes generalized non-coplanar geometry per Table 4-2)
! ==========================================================
!
! At the cg:
!  to Bacc1
! d0x1 = 0.203
! d0x2 = 0.0
! d0x3 = 0.0
!
! to Bacc2
! d0y1 = 0.0
! d0y2 = 0.203
! d0y3 = 0.0
!
! to Bacc3
! d0z1 = 0.0
! d0z2 = 0.0
! d0z3 = 0.203
!
! On the x arm, using location of bacc6 for reference, rx = 2.403:
! to Bacc7
! d1y1 = -.203
! d1y2 = 0.0
! d 1y3 = 0.203
!
! to Bacc6
! d1z1 = 0.0
! d1z2 = 0.0
! d1z3 = 0.0
!
! On the y arm, using location of bacc4 for reference, ry = 2.103:
! to Bacc5
! d2x1 = 0.0
! d2x2 = -.203
! d2x3 = .203
!
! to Bacc4
! d2z1 = 0.0
! d2z2 = 0.0
```

```
! d2z3 = 0.0
!
! On the z arm, using z axis as reference position, rz = 3.20:
! to Bacc8
! d3x1 = 0.203
! d3x2 = 0.0
! d3x3 = 0.0
!
! to Bacc9
! d3y1 = 0.0
! d3y2 = 0.203
! d3y3 = 0.0
!
! End general non-coplanar geometry data
! ====================================

! ====================================
! Evaluate compensation coefficients:
! ====================================
!
a4 = -(d2z3 - d0z3)/(2.*ry) + (d3y2 - d0y2)/(2.*rz)
a5 = -(d2z3 - d0z3)/(2.*ry)

b4 =  (d1z3 - d0z3)/(2.*rx)
b5 = -(d3x1 - d0x1)/(2.*rz) + (d1z3 - d0z3)/(2.*rx)

c1 = -(d1y3 - d0y3)/(2.*rx)
c2 = -(d2x3 - d0x3)/(2.*ry)
c3 =  (d1y1 - d0y1)/(2.*rx) + (d2x2 - d0x2)/(2.*ry)
c4 = -(d1y2 - d0y2)/(2.*rx)
c5 =  (d2x1 - d0x1)/(2.*ry)
c6 = -(d1y2 - d0y2)/(2.*rx) + (d2x1 - d0x1)/(2.*ry)
c7 =  (d1y1 - d0y1)/(2.*rx) - (d2x2 - d0x2)/(2.*ry)
c8 = -(d2x3 - d0x3)/(2.*ry)
c9 =  (d1y3 - d0y3)/(2.*rx)

! ================================================================
! Define coefficient matrix to compensate for non-ideal geometry
! ================================================================

Ai(1,1) = 1.0
Ai(2,1) = 0.0
Ai(3,1) = -c1/(1.0 + c3)
```

```
Ai(1,2) = 0.0
Ai(2,2) = 1.0
Ai(3,2) = -c2/(1.0 + c3)

Ai(1,3) = 0.0
Ai(2,3) = 0.0
Ai(3,3) = 1.0/(1.0 + c3)

! ================
! Initialize T matrix
! ================

t(1,1) = +cos(theta)*cos(psi)
t(2,1) = +sin(phi)*sin(theta)*cos(psi) + cos(phi)*sin(psi)
t(3,1) = -cos(phi)*sin(theta)*cos(psi) + sin(phi)*sin(psi)

t(1,2) = -cos(theta)*sin(psi)
t(2,2) = -sin(phi)*sin(theta)*sin(psi) + cos(phi)*cos(psi)
t(3,2) = +cos(phi)*sin(theta)*sin(psi) + sin(phi)*cos(psi)

t(1,3) = +sin(theta)
t(2,3) = -sin(phi)*cos(theta)
t(3,3) = +cos(phi)*cos(theta)
! ========================================
! Set flag for large incremental angular velocities
! ========================================

nflag2 = 0

! ================================
! Initialize body angular velocities
! ================================

bomegx = 0.
bomegy = 0.
bomegz = 0.

! Determine the number of points (for controlling iterative
! calculations)
!
npts = (ub-lb)+1
do 955 i = 1,npts
```

```
      !=================================
      ! Estimated body angular accelerations
      !=================================

      bomegdx(i) = ((bacc(4).y(i)-bacc(3).y(i))/ry
     *          + (bacc(2).y(i)-bacc(9).y(i))/rz)*HGacc
      bomegdy(i) = ((bacc(3).y(i)-bacc(6).y(i))/rx
     *          + (bacc(8).y(i)-bacc(1).y(i))/rz)*HGacc
      bomegdz(i) = ((bacc(7).y(i)-bacc(2).y(i))/rx
     *          + (bacc(1).y(i)-bacc(5).y(i))/ry)*HGacc

      !=================================
      ! First order compensation coefficients
      !=================================
      !
      Q(1) = a4*bomegx**2 + a5*bomegy**2
      Q(2) = b4*bomegx**2 + b5*bomegy**2
      Q(3) = c4*bomegx**2 + c5*bomegy**2  + c6*bomegz**2 +
     *     c7*bomegx*bomegy + c8*bomegx*bomegz + c9*bomegy*bomegz

      R(1) = bomegdx(i) - Q(1)
      R(2) = bomegdy(i) - Q(2)
      R(3) = bomegdz(i) - Q(3)

      !=================================================
      ! Angular accelerations compensated for non-coplanar geom
      !=================================================

      bomegdx(i) = Ai(1,1)*R(1) + Ai(1,2)*R(2) + Ai(1,3)*R(3)
      bomegdy(i) = Ai(2,1)*R(1) + Ai(2,2)*R(2) + Ai(2,3)*R(3)
      bomegdz(i) = Ai(3,1)*R(1) + Ai(3,2)*R(2) + Ai(3,3)*R(3)

      bomegx = bomegx + bomegdx(i)*del
      bomegy = bomegy + bomegdy(i)*del
      bomegz = bomegz + bomegdz(i)*del

      bomegxx(i) = bomegx
      bomegyy(i) = bomegy
      bomegzz(i) = bomegz

      ! Constant angular velocities for testing T matrix.
      ! bomegx = 0.         bomegy = 0.  bomegz = 0.
```

```
! ==========================================
! Update body axes relative to fixed coordinates
! ==========================================

call trans(del,bomegx,bomegy,bomegz,omeg,T,TDT,nflag,nflag2,nstep)

! ================================================
! Compensate for non-centroidal location of accelerometers
! ================================================

bacc(1).y(i) = bacc(1).y(i) + d0x1*(bomegz**2 + bomegy**2)/Gacc
bacc(2).y(i) = bacc(2).y(i) + d0y2*(bomegx**2 + bomegz**2)/Gacc
bacc(3).y(i) = bacc(3).y(i) + d0z3*(bomegx**2 + bomegy**2)/Gacc

if (nflag .eq. 1) goto 981

! Translational accelerations transformed to fixed coordinate system
!
faccx(i) = T(1,1)*bacc(1).y(i) + T(1,2)*bacc(2).y(i)
*          + T(1,3)*bacc(3).y(i)
faccy(i) = T(2,1)*bacc(1).y(i) + T(2,2)*bacc(2).y(i)
*          + T(2,3)*bacc(3).y(i)
faccz(i) = T(3,1)*bacc(1).y(i) + T(3,2)*bacc(2).y(i)
*          + T(3,3)*bacc(3).y(i)
!
! Angular accelerations transformed to fixed coordinate system
!
fomegdx(i) = T(1,1)*bomegdx(i)
*          +T(1,2)*bomegdy(i)+T(1,3)*bomegdz(i)
fomegdy(i) = T(2,1)*bomegdx(i)
*          +T(2,2)*bomegdy(i)+T(2,3)*bomegdz(i)
fomegdz(i) = T(3,1)*bomegdx(i)
*          +T(3,2)*bomegdy(i)+T(3,3)*bomegdz(i)
!
! Angular velocities transformed to fixed coordinate system
!
fomegx(i) = T(1,1)*bomegxx(i)
*          +T(1,2)*bomegyy(i)+T(1,3)*bomegzz(i)
fomegy(i) = T(2,1)*bomegxx(i)
*          +T(2,2)*bomegyy(i)+T(2,3)*bomegzz(i)
fomegz(i) = T(3,1)*bomegxx(i)
*          +T(3,2)*bomegyy(i)+T(3,3)*bomegzz(i)
```

```
      time = i*del
      !
      ! Calculate angular displacements from direction cosine matrix:
      !       (Rotations up to 90 degrees)

      rtd = 57.2958

      if (abs(T(3,1)) .gt. .9999999) go to 953

      beta(i) = -asin(T(3,1))

      cosb = cos(beta(i))

      if (cosb .eq. 0.) go to 981
      if (abs(T(3,2)/cosb) .gt. 1.) go to 981
      if (abs(T(2,1)/cosb) .gt. 1.) go to 981
      alpha(i) = asin(T(3,2)/cosb)

      gamma(i) = asin(T(2,1)/cosb)

      goto 954

953   alpha(i) = acos(T(2,2))

      arg1 = 1.570796327
      arg2 = -T(3,1)

      beta(i) = sign(arg1,arg2)

      gamma(i) = 0.

954   continue

      alpha(i) = alpha(i)*rtd
      beta(i) = beta(i)*rtd
      gamma(i) = gamma(i)*rtd

      ang1(i) = acosd(T(1,1))
      ang2(i) = acosd(T(2,2))
      ang3(i) = acosd(T(3,3))
```

```
955   continue
      ! ==========================================================
      ! Section for adding +1G body force to Z axis acceleration
      !       if required
      ! ==========================================================
      !
      ! do 199 i=lb,ub
      !       faccz(i) = faccz(i) + 1.0
      ! 199   continue
      !
      ! Compute translational velocities (fixed coordinates)
      !  Hgacc = 193.04 for in/sec
      ! For initial results, use in/sec ...
      !
      inivel = 0.0

      call trapint(faccx,fvelx,1,npts,inivel,HGacc*del)
      call trapint(faccy,fvely,1,npts,inivel,HGacc*del)
      call trapint(faccz,fvelz,1,npts,inivel,HGacc*del)

986   continue
      ! ========================================
      ! End coordinate transformation computations
      ! Go to output section
      ! ========================================
      !
      ! Write output files:
      !
      go to 985
981   write (6,982)
982   format (/8x,'ACOS argument greater than 1. Results truncated')
983   write (6,984)
984   format (/4x,'Cosine of Beta = 0. Results truncated')

985   continue
      !
      ! ======================================
      ! Begin writting output data, assign filenames
      ! and reset header variables as required
      ! ======================================
      !
```

```
! Set control values for output - In general, output file
! should contain information of output quantity, axis, body
! or fixed coordinates, and units (generally English).
! UDS control structure
!
uds_c.filever = 'UDS-1992'
uds_c.fileform = 'Y'
uds_c.chanform = 'Y'
uds_c.specs_only = .false.
uds_c.dimsys = 'ENG'
!
! Set template UDS object's elements to appropriate values
!
! Write accelerations relative to fixed coordinates:
!
template.units = 'G"S'
template.dattyp = 'ACCLERATION'
template.status = 'FIXED_COORDS'

outfile(1) = 'F'//testno//'AM00.00X'
outfile(2) = 'F'//testno//'AM00.00Y'
outfile(3) = 'F'//testno//'AM00.00Z'

!
! For each curve written, copy signal data from local array to
! the output UDS object, and then write object to the UDS file
!
template.axis = 'XF'
call copy_array(faccx,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(1),template,uds_c,ios)
template.axis = 'YF'
call copy_array(faccy,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(2),template,uds_c,ios)
template.axis = 'ZF'
call copy_array(faccz,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(3),template,uds_c,ios)
!
! Write body angular velocities:
!
template.units = 'RADIANS/SEC'
template.dattyp = 'ANGULAR VELOCITY'
template.status = 'BODY_COORDS'
```

```
outfile(4) = 'B'//testno//'ROTV.00X'
outfile(5) = 'B'//testno//'ROTV.00Y'
outfile(6) = 'B'//testno//'ROTV.00Z'

template.axis = 'XB'
call copy_array(bomegxx,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(4),template,uds_c,ios)
template.axis = 'YB'
call copy_array(bomegyy,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(5),template,uds_c,ios)
template.axis = 'ZB'
call copy_array(bomegzz,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(6),template,uds_c,ios)
!
! Write body angular accelerations:
!
template.units = 'RADIANS/SEC'
template.dattyp = 'ANGULAR ACCLERATION'
template.status = 'BODY_COORDS'

outfile(7) = 'F'//testno//'ROTA.00X'
outfile(8) = 'F'//testno//'ROTA.00Y'
outfile(9) = 'F'//testno//'ROTA.00Z'

template.axis = 'XB'
call copy_array(fomegdx,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(7),template,uds_c,ios)
template.axis = 'YB'
call copy_array(fomegdy,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(8),template,uds_c,ios)
template.axis = 'ZB'
call copy_array(fomegdz,bot,top,template.y,nfp_l,nlp_u,lb,ub)
call uds_write(outfile(9),template,uds_c,ios)
!
! Write direction cosine data:
!
template.units = 'DEGREES'
template.dattyp = 'AXIS ORIENTATION'
template.status = 'BODY_TO_FIXED'

outfile(10) = 'F'//testno//'DCOS.011'
```

```
outfile(11) = 'F'//testno//'DCOS.022'
outfile(12) = 'F'//testno//'DCOS.033'

template.axis = 'XU'
call copy_array(ang1,bot,top,template.y,nfp_1,nlp_u,lb,ub)
call uds_write(outfile(10),template,uds_c,ios)
template.axis = 'YV'
call copy_array(ang2,bot,top,template.y,nfp_1,nlp_u,lb,ub)
call uds_write(outfile(11),template,uds_c,ios)
template.axis = 'ZW'
call copy_array(ang3,bot,top,template.y,nfp_1,nlp_u,lb,ub)
call uds_write(outfile(12),template,uds_c,ios)
!
! Write angular displacements (fixed coords) data:
!
template.units = 'DEGREES'
template.dattyp = 'ANGULAR DISPLACEMENT'
template.status = 'FIXED_COORD'

outfile(13) = 'F'//testno//'ROTD.00X'
outfile(14) = 'F'//testno//'ROTD.00Y'
outfile(15) = 'F'//testno//'ROTD.00Z'

template.axis = 'XF'
call copy_array(alpha,bot,top,template.y,nfp_1,nlp_u,lb,ub)
call uds_write(outfile(13),template,uds_c,ios)
template.axis = 'YF'
call copy_array(beta,bot,top,template.y,nfp_1,nlp_u,lb,ub)
call uds_write(outfile(14),template,uds_c,ios)
template.axis = 'ZF'
call copy_array(gamma,bot,top,template.y,nfp_1,nlp_u,lb,ub)
call uds_write(outfile(15),template,uds_c,ios)
!
! Write angular velocities (fixed coords):
!
template.units = 'RADIANS/SEC'
template.dattyp = 'ANGULAR VELOCITY'
template.status = 'FIXED_COORD'

outfile(16) = 'F'//testno//'ROTV.00X'
outfile(17) = 'F'//testno//'ROTV.00Y'
outfile(18) = 'F'//testno//'ROTV.00Z'
```

```
      template.axis = 'XF'
      call copy_array(fomegx,bot,top,template.y,nfp_l,nlp_u,lb,ub)
      call uds_write(outfile(16),template,uds_c,ios)
      template.axis = 'YF'
      call copy_array(fomegy,bot,top,template.y,nfp_l,nlp_u,lb,ub)
      call uds_write(outfile(17),template,uds_c,ios)
      template.axis = 'ZF'
      call copy_array(fomegz,bot,top,template.y,nfp_l,nlp_u,lb,ub)
      call uds_write(outfile(18),template,uds_c,ios)
      !
      !-----------------------------------------------
      !
      ! Write translational velocities (fixed coords):
      !
      template.units = 'IN/SEC'
      template.dattyp = 'VELOCITY'
      template.status = 'FIXED_COORD'

      outfile(19) = 'F'//testno//'VM00.00X'
      outfile(20) = 'F'//testno//'VM00.00Y'
      outfile(21) = 'F'//testno//'VM00.00Z'

      template.axis = 'XF'
      call copy_array(fvelx,bot,top,template.y,nfp_l,nlp_u,lb,ub)
      call uds_write(outfile(19),template,uds_c,ios)
      template.axis = 'YF'
      call copy_array(fvely,bot,top,template.y,nfp_l,nlp_u,lb,ub)
      call uds_write(outfile(20),template,uds_c,ios)
      template.axis = 'ZF'
      call copy_array(fvelz,bot,top,template.y,nfp_l,nlp_u,lb,ub)
      call uds_write(outfile(21),template,uds_c,ios)
      !
      ! Report completion of tasks
      !
      write (6,700)
  700 format (/6X,'The Following Files Have Been Created:',/,
     * 3X,'Translational Accelerations - Inertial Coordinates.')
      write (6,707) outfile(1),outfile(2),outfile(3)
      write (6,701)
  701 format (/3X,'Rotational Velocities - Body Coordinates')
      write (6,707) outfile(4),outfile(5),outfile(6)
```

```
      write (6,702)
702   format (/3X,'Rotational Accelerations - Inertial Coordinates')
      write (6,707) outfile(7),outfile(8),outfile(9)
      write (6,703)
703   format (/3X,'Direction Cosines Relating Principal Axes')
      write (6,707) outfile(10),outfile(11),outfile(12)
      write (6,704)
704   format (/3X,'Rotational Displacements - Inertial Coordinates')
      write (6,707) outfile(13),outfile(14),outfile(15)
      write (6,705)
705   format (/3X,'Rotational Velocities - Inertial Coordinates')
      write (6,707) outfile(16),outfile(17),outfile(18)
      write (6,706)
706   format (/3X,'Translational Velocities - Inertial Coordinates')
      write (6,707) outfile(19),outfile(20),outfile(21)
      write (6,708)
708   format (3X,'Load Curves Written to: LC_FILE.DAT')

707   format(3x,3(5x,a))
      !
      ! ================================
      ! Finished writting UDS output files
      ! ================================
      !================================================
      ! Begin writting load curve data (three translational
      ! velocities and three rotational velocities about
      ! the center of gravity, to data file, LC_FILE.DAT
      ! Units are English (in-lb-sec) for compatibility
      ! with anatomic model.
      !================================================
      !
      ! Output data interval:
      !
      do 500 j=1,npts
            timeout(j) = (j-1)*del
      !
      ! write(6,*)j,del,timeout(j)
      !
500   continue

      open(unit=20, access='sequential', status='new',
     * file='lc_file.dat')
      !
```

```
      ! write data for x velocity load curve
      !
     write(20, 7301),npts

          do 501 j=1,npts
     write(20, 6100) timeout(j),fvelx(j)
501   continue
      !
      ! write data for y velocity load curve
      !
     write(20, 7302),npts

          do 502 j=1,npts
     write(20, 6100) timeout(j),fvely(j)
502   continue
      !
      ! write data for z velocity load curve
      !
     write(20, 7303),npts

          do 503 j=1,npts
     write(20, 6100) timeout(j),fvelz(j)
503   continue
      !
      ! write data for x-axis angular velocity load curve
      !
     write(20, 7304),npts

          do 504 j=1,npts
     write(20, 6100) timeout(j),fomegx(j)
504   continue
      !
      ! write data for y-axis angular velocity load curve
      !
     write(20, 7305),npts

          do 505 j=1,npts
     write(20, 6100) timeout(j),fomegy(j)
505   continue
      !
      ! write data for z-axis angular velocity load curve
      !
```

```
      write(20, 7306),npts

          do 506 j=1,npts
      write(20, 6100) timeout(j),fomegz(j)
506   continue

      close(unit=20, status='keep')
      !
      ! Note that breaking string constants in format statements
      ! or elsewhere across lines is NOT recommended - it may
      ! be (at best) non-portable and (at worst) illegal
      !
6100  format(2(3x, 1pe10.3))
7301  format('c load curve for x velocity',/,1x,'lcd',3x,'1',3x,i4)
7302  format('c load curve for y velocity',/,1x,'lcd',3x,'2',3x,i4)
7303  format('c load curve for z velocity',/,1x,'lcd',3x,'3',3x,i4)
7304  format('c load curve for angular vel. about x axis',
     *        /,1x,'lcd',3x,'4',3x,i4)
7305  format('c load curve for angular vel. about y axis',
     *        /,1x,'lcd',3x,'5',3x,i4)
7306  format('c load curve for angular vel. about z axis',
     *        /,1x,'lcd',3x,'6',3x,i4)

      close(unit=20, status='keep')
      !
      ! ================================
      ! Finished writting load curve data
      ! ================================
      !
990   continue

      end
      !
      ! ========================================
      ! Coordinate transformation subroutine.  From
      ! algorithm developed by H. Weinstock, DTS-76.
      ! ========================================
      !
      subroutine TRANS(del,bomegx,bomegy,bomegz,omeg,T,TDT,
     *                 nflag,nflag2,nstep)

      dimension omeg(3,3),T(3,3),TDT(3,3)
```

```
      double precision T,TDT

      do 100 i = 1,3
      omeg(i,i) = 0.
100   continue

      omeg(1,2) = -bomegz
      omeg(1,3) =  bomegy
      omeg(2,3) = -bomegx
      omeg(2,1) = -omeg(1,2)
      omeg(3,1) = -omeg(1,3)
      omeg(3,2) = -omeg(2,3)

      bomegv = sqrt(bomegx**2 + bomegy**2 + bomegz**2)

      if (bomegv*del .gt. .001) then
            nflag2 = 2
            ninc = bomegv*del/.001
            nstep = int(ninc) + 1
            delt = del/nstep
      else
            delt = del
            nstep = 1
      end if

      do 120  n = 1,nstep
      do 110  i = 1,3
      do 110  j = 1,3
      TDT(i,j) = 0.
      do 110 k = 1,3
      TDT(i,j) = TDT(i,j) + T(i,k)*omeg(k,j)
110   continue

      do 120 i = 1,3
      do 120 j = 1,3
      nflag = 0
      T(i,j) = T(i,j) + TDT(i,j)*delt
      if (T(i,j) .gt.  1.) T(i,j) =  1.
      if (T(i,j) .lt. -1.) T(i,j) = -1
      if (T(i,j) .gt. 1. .or. T(i,j) .lt. -1.) nflag = 1
120   continue
```

```
      return
      end
      !
      ! ============================
      ! TRAPINT: trapezoidal integration
      ! ============================
      !
      subroutine trapint(z,w,nfp,nlp,w0,wfac)

      dimension z(-1000:*),w(-1000:*)
      double precision zm,zi,wm,wi
      integer nfp, nlp

      w(0)=w0

      zm=z(0)
      wm=w0

      do 10 i=1,nlp
        zi=z(i)
        wi=wm+wfac*(zm+zi)
        w(i)=wi
        zm=zi
        wm=wi
10    continue

      zm=z(0)
      wm=w0

      do 20 i=-1,nfp,-1
        zi=z(i)
        wi=wm-wfac*(zm+zi)
        w(i)=wi
        zm=zi
        wm=wi
20    continue

      return

      end
      !
```

```
! ========================================
! NHTSA UDS routine for preparing output data
! ========================================
! COPY_ARRAY: Load a source array into a destination array,bot,top,
! using a range in the source array specified by a lower
! and upper bound
!
subroutine copy_array(source,sfp,slp,dest,dfp,dlp,lb,ub)
!
integer         sfp,slp,dfp,dlp,lb,ub
real*4 source(sfp:slp),dest(dfp:dlp)
!
integer i

do i=lb,ub
        dest(i) = source(i)
enddo

return

end
```

REFERENCES

1. Society of Automotive Engineers. Information Report J885, "Human Tolerance to Impact Conditions as Related to Motor Vehicle Design." *1994 SAE Handbook.* Volume 3 - On-Highway Vehicle and Off-Highway Machinery.

2. F. A. Bandak and R. H. Eppinger. *A 3-D Finite Element Analysis of the Human Brain Under Combined Rotational and Translational Acceleration,* 38th Stapp Car Crash Conference Proceedings, October 31-November 4, 1994, Ft. Lauderdale, FL. SAE Paper No. 942215.

3. DiMasi, Frank. *3-D Anatomic Brain Model for Relating Cortical Strains to Automobile Crash Loadings.* US/DOT/Volpe Center and R. Eppinger and J. Marcus US/DOT/NHTSA, 13th International Experimental Safety Vehicle Conference, November 4-7, Paris France, 1991, Paper No 91-S8-O-11.

4. Plank, G., Weinstock, H., Coltman, M. and H. S. Lee. *Methodology for the Calibration of and Data Acquisition with a Six Degree-of-Freedom Acceleration Measurement Device,* Report No. DOT-HS-807-370 (DOT-TSC-NHTSA-88-3), June 1989.

5. Broxmeyer, Charles. *Inertial Navigation Systems.* McGraw Hill Electronic Sciences Series, McGraw Hill Book Co., New York 1964.

6. Stillman, D.W. and J. O. Hallquist. *LS-INGRID: A Pre-Processor And Three-Dimensional Mesh Generator for the Programs LS-DYNA3D, LS-NIKE3D, and TOPAZ3D.* Livermore Software Technology Corporation, Livermore CA, LSTC Report 1019, March 1993.

7. Hallquist, J. O., Stillman, D. W. and T. S. Lin. *LS-DYNA3D User's Manual.* Livermore Software Technology Corporation, Livermore CA, LSTC Report 1007, January 1993.

8. Hallquist, J. O. *LS-TAURUS An Interactive Post-Processor for the Analysis Codes LS-NIKE3D, LS-DYNA3D, LS-NIKE3D, and TOPAZ3D.* Livermore Software Technology Corporation, Livermore CA, LSTC Report 1001, April 1992.

9. Laughlin, D. R. *A Magnetohydrodynamic Angular Motion Sensor for Anthropomorphic Test Device Instrumentation* Applied Technology Associates, Albuquerque, NM. Proceedings of the 33 Stapp Car Crash Conference, October 4-6, 1989, Washington, DC. Stapp Conference Paper No. 892428.

•U.S. GOVERNMENT PRINTING OFFICE: 1995-602-421/20014